Universidade do Minho

Universidade de Vigo

linguaMATICA

Volume 16, Número 2 (2024)

ISSN: 1647-0818

Volume 16, Número 2 – 2024

LinguaMÁTICA

ISSN: 1647–0818

Editores Executivos

Marcos Garcia
Hugo Gonçalo Oliveira

Editores

Alberto Simões
José João Almeida
Xavier Gómez Guinovart

Conteúdo

Artigos de Investigação

Detecção Precoce de Transtornos de Saúde Mental em Português
Bruno Nagamatu & Ivandré Paraboni 3

Artigos Técnicos

Aperfeiçoando a Hifenização Automática em Português no TEX
Leonardo Araujo & Aline Benevides 23

PROPOR 2024 | Artigos Convidados

Explorando a Eficácia dos Modelos Generativos em Tarefas de Análise de Sentimentos no Português Brasileiro
Gladson de Araújo, Tiago de Melo & Carlos Figueiredo 41

Detecção de Posicionamentos a partir de Textos usando Expansão Semiautomática de Córpus
Camila Pereira & Ivandré Paraboni 59

MultiWOZ-PT: Um Conjunto de Diálogos Orientados a Tarefas em Português
Patrícia Ferreira, Francisco Pais, Catarina Silva, Ana Alves & Hugo Gonçalo Oliveira 75

Explorando Técnicas de Aprendizado em Modelos de Linguagem para Classificação de Discurso de Ódio e Ofensivo em Português
G. Assis, A. Amorim, J. Carvalho, M. Ferro, D. de Oliveira, D. Vianna & A. Paes 91

Avaliação Automática do Nível de Complexidade de Textos em Português Europeu
Eugénio Ribeiro, Nuno Mamede & Jorge Baptista 115

Reconhecimento de Entidades Nomeadas e Vazamento de Dados em Textos Legislativos: Uma Reavaliação da Literatura
Rafael Nunes, André Spritzer, Carla Freitas & Dennis Balreira 141

Extração de Informação Aberta com LLM para a Língua Portuguesa
Bruno Cabral, Marlo Souza & Daniela Barreiro Claro 167

RoBERTaLexPT:
um modelo RoBERTa jurídico pré-treinado com deduplicação para língua Portuguesa
Eduardo Garcia et al. 183

Rotulação e Caracterização de Conteúdo Tóxico
de Comunidades do Reddit no Brasil
Luiz Quevedo, Ana Pagano, Adriana Pagano & Ana Silva 201

Editorial

Finalizamos este 2024 com algumas novidades na revista Linguamática. Neste número, para além de artigos de investigação e técnicos de carácter regular, este número da revista Linguamática inclui uma secção dedicada a trabalhos selecionados da 16th International Conference on the Computational Processing of the Portuguese Language *(PROPOR 2024), o principal encontro científico da área das tecnologias da linguagem e fala para a língua portuguesa. A edição de 2024, a primeira num país que não Portugal ou Brasil, realizou-se na Universidade de Santiago de Compostela, na Galiza, entre os dias 12 e 15 de março de 2024.*

Os artigos foram selecionados pelos editores da Linguamática e resultam de um convite feito aos autores, onde lhes foi proposto descreverem o seu trabalho num artigo mais extenso, em português, e com novidades relativamente aos artigos originais. Apesar dos convites, os novos artigos passaram pelo normal processo de revisão. Como em alguns casos, este processo ainda decorre, alguns destes artigos serão publicados no próximo número da Linguamática.

Os últimos meses marcaram ainda uma novidade na revista Linguamática. Dado o ritmo com que o estado da arte das tecnologias da linguagem tem avançado, acreditamos que é importante que os artigos fiquem disponíveis para a comunidade o mais depressa possível, sem esperar pelo próximo número da revista, que sai apenas de seis em seis meses. Assim, inaugurámos uma secção da página da Linguamática onde disponibilizamos, em versão "pré-impressa," os artigos que vão sendo aceites e que já estão prontos a ser publicados no próximo número.

Esperamos, deste modo, continuar a contribuir para que os avanços na investigação do processamento das línguas ibéricas alcancem a visibilidade e a divulgação que merecem.

Hugo Gonçalo Oliveira
Marcos Garcia

Revisão

A comissão científica da **Lingua**MÁTICA pode ser consultada na página Web da revista, em https://linguamatica.com/index.php/linguamatica/about/editorialTeam.

Para esta edição, colaboraram os seguintes investigadores:

- **Alexandre Rademaker**, IBM Research & EMAp/FGV
- **Aline Paes**, Universidade Federal Fluminense
- **Álvaro Iriarte Sanroman**, Universidade do Minho
- **Amália Mendes**, Universidade de Lisboa
- **Ana Isabel Mata**, Faculdade de Letras da Universidade de Lisboa
- **Ana Oliveira Alves**, Universidade de Coimbra
- **Ana Rita Sá-Leite**, Georg-August-Universität Göttingen
- **Anastasiia Ogneva**, UiT The Arctic University of Norway
- **Catarina Oliveira**, Universidade de Aveiro
- **Cláudia Freitas**, Pontifícia Universidade Católica do Rio de Janeiro
- **Cristiane Namiuti**, Universidade Estadual do Sudoeste da Bahia
- **Denilson Alves Pereira**, Universidade Federal de Lavras
- **Doaa Samy Khalil Shawer**, Universidad Complutense de Madrid
- **Eloize Seno**, Instituto Federal de São Paulo
- **Eugénio Ribeiro**, INESC-ID e Instituto Universitário de Lisboa (ISCTE-IUL)
- **Hidelberg Oliveira Albuquerque**, Universidade Federal Rural de Pernambuco
- **Ivandré Paraboni**, Universidade de São Paulo
- **Jorge Baptista**, Universidade do Algarve
- **José João Almeida**, Universidade do Minho
- **José Ramom Pichel Campos**, Universidade de Santiago de Compostela
- **Irene Renau**, Pontificia Universidad Católica de Valparaíso
- **Irene Rodrigues**, Universidade de Évora
- **Leonardo Zílio**, Friedrich-Alexander-Universität Erlangen-Nürnberg
- **Marcos Didonet Del Fabro**, Universite Paris-Saclay
- **Marlo Souza**, Universidade Federal da Bahia
- **Maria José Finatto**, Universidade Federal do Rio Grande do Sul
- **Maria das Graças Volpe Nunes**, Universidade de São Paulo
- **Nuno Guimarães**, INESC-TEC e Universidade do Porto
- **Pablo Faria**, Universidade de Campinas
- **Pablo Gamallo**, Universidade de Santiago de Compostela
- **Rafael Anchiêta**, Instituto Federal do Piauí
- **Renata Vieira**, Universidade de Évora
- **Ricardo Rodrigues**, Universidade de Coimbra
- **Sandra Maria Aluísio**, Universidade de São Paulo
- **Sara Carvalho**, Universidade de Aveiro
- **Thiago Pardo**, Universidade de São Paulo
- **Valeria Delisandra Feltrim**, Universidade Estadual de Maringá
- **Violeta Quental**, Pontifícia Universidade Católica do Rio de Janeiro

Artigos de Investigação

Proposta recebida em Agosto 2024 e aceite para publicação em Novembro 2024.

Detecção precoce de transtornos de saúde mental em Português

Portuguese mental health early risk prediction

Bruno Issamo Tagava Nagamatu
Escola de Artes, Ciências e Humanidades - Universidade de São Paulo

Ivandré Paraboni
Escola de Artes, Ciências e Humanidades - Universidade de São Paulo

Resumo

O presente estudo enfoca o problema da detecção precoce de transtornos de saúde mental nos moldes da série de desafios *eRisk* (originalmente voltado ao domínio de fóruns de discussão sobre saúde mental no idioma inglês) em uma rede social de propósito geral em português. De forma mais específica, propõe a adaptação de uma estratégia vencedora em diversas edições desse *shared task* para o caso da detecção precoce de depressão e transtorno de ansiedade no domínio do Twitter/X brasileiro, usando para esse fim uma abordagem inédita baseada em LLMs com uso de engenharia de *prompts*. Os resultados obtidos indicam que o uso de LLMs apresenta maior poder de antecipação de diagnóstico em relação a abordagens tradicionais da área, e que a detecção com base em publicações da redes social de propósito geral é potencialmente mais desafiadora do que na formulação original do problema, sendo dependente da proximidade das mensagens do momento do diagnóstico na ordem cronológica da *timeline* do *Twitter*/X.

Palavras chave

saúde mental; depressão; ansiedade; detecção precoce; redes sociais

Abstract

The present study focuses on the early detection of mental health disorders along the lines of the eRisk shared task series (originally devoted to online mental health discussion domain in the English language) in general-purpose Portuguese social media. More specifically, we adapt a strategy that has won a number of shared tasks to the case of early detection of depression and anxiety disorders in the Brazilian Twitter/X domain, using for this purpose a novel approach based on LLMs and prompt engineering. Our results indicate that the use of LLMs affords greater power to anticipate diagnosis if compared to traditional approaches in the field, and that detection based on general-purpose social media text is potentially more challenging than in the original problem formulation, being dependent on the proximity of the messages to the moment of diagnosis in the chronological order of the timeline on Twitter/X.

Keywords

mental health; depression; anxiety; early risk prediction; social media

1. Introdução

Transtornos de saúde mental como depressão, ansiedade, bipolaridade, distúrbios alimentares, esquizofrenia e outros atingem cerca de 10,7% da população mundial (Dattani et al., 2021). Nesse contexto, o aumento da popularidade das mídias sociais fez com que esse meio se tornasse uma rica fonte de dados para pesquisadores da área de Processamento de Línguas Naturais (PLN). Em estudos desse tipo, textos provenientes de redes sociais são comumente utilizados para detecção de transtornos de saúde mental com uso de métodos de aprendizado de máquina supervisionados (Zafar & Chitnis, 2020; Chiong et al., 2021; Amanat et al., 2022; Chen et al., 2023; Lestandy et al., 2024), constituindo uma tarefa que pode ser vista como um caso particular de caracterização autoral ou *author profiling* (da Silva et al., 2020; Bevendorff et al., 2022; Flores et al., 2022; Pavan et al., 2023).

Para fins de treinamento e teste, modelos tradicionais de detecção de transtornos de saúde mental a partir de texto tipicamente fazem uso da totalidade dos dados (e.g., *timelines*) disponíveis na forma de um córpus rotulado. A tarefa computacional assume assim a forma de um problema de análise da rede social em que, dada uma *timeline* de interesse, o objetivo é decidir se aquele conjunto de publicações corresponde, por exemplo, a um perfil depressivo ou não.

Apesar das evidentes aplicações práticas de abordagens desse tipo, entretanto, estudos como

DOI: 10.21814/lm.16.2.440
This work is Licensed under a Creative Commons Attribution 4.0 License

os participantes da série de desafios (ou *shared tasks*) *eRisk* (Losada et al., 2017, 2018, 2019; Parapar et al., 2022, 2023) introduziram o conceito de detecção *precoce* de transtornos de saúde mental, na qual o objetivo passa a ser a detecção da forma mais antecipada possível, ou seja, com base no menor volume possível de publicações seguindo a ordem cronológica em que são apresentadas na rede social.

Aplicações voltadas à detecção precoce são mais associadas à prevenção de transtornos do que à detecção de transtornos já diagnosticados, o que pode apresentar certas vantagens de natureza prática. Por exemplo, podem agilizar o suporte a indivíduos que manifestam os primeiros sinais de um transtorno grave, ou até mesmo auxiliar na sua prevenção (Parapar et al., 2023). Entretanto, observa-se que as tarefas tratadas nos desafios *eRisk* são quase que exclusivamente baseadas em córpus do gênero *Reddit* (e.g., discussões *online* sobre saúde mental, ou outros tópicos específicos) cujos dados são de natureza distinta de uma rede social de propósito geral, ou seja, que discutem potencialmente qualquer tipo de assunto de forma irrestrita.

Um exemplo típico de rede social dita de propósito geral, e que é o objeto de estudo da presente pesquisa, é o *microblog* *Twitter*/X. Em redes sociais desse tipo, conforme será discutido na seção 4, observa-se que a maior parte das publicações tende a não ter relação direta com o estado de saúde mental do usuário autor destas publicações, o que configura um problema computacional bastante distinto do considerado na formulação original do problema da série de desafios *eRisk*, em que o conteúdo textual trata exclusivamente do transtorno a ser detectado (e.g, depressão).

Além disso, cabe observar também que os estudos da série *eRisk* são voltados ao idioma inglês, e apenas recentemente começam a ser exploradas em iniciativas correlatas desenvolvidas para a língua espanhola (Mármol-Romero et al., 2023). No caso específico da língua portuguesa, por outro lado, embora existam iniciativas baseadas na formulação tradicional do problema (Mann et al., 2020; de Souza et al., 2022; dos Santos et al., 2023; dos Santos & Paraboni, 2023, 2024) e outras (de Cássia Alves et al., 2024; Mendes & Caseli, 2024), não há, até onde temos conhecimento, estudos de detecção precoce de transtornos de saúde mental baseados em dados em português.

Finalmente, observa-se ainda que, pelo caráter competitivo de desafios como *eRisk* e similares, e suas restrição de tempo para entrega de resultados, estudos existentes tendem a privilegiar métodos computacionais mais tradicionais e de menor custo computacional em detrimento de alternativas mais robustas, porém com custo computacional geralmente superior. Esse é o caso, por exemplo, do uso de grandes modelos de língua (LLMs) que, embora tenham começado a ser aplicados à formulação tradicional do problema de detecção de transtornos de saúde mental (Tao et al., 2023; Agrawal, 2024), ainda não figuram entre as melhores soluções existentes para detecção precoce na série de desafios *eRisk* e estudos correlatos.

Com base nestas observações, o presente estudo enfoca o problema da detecção precoce de transtornos de saúde mental em uma rede social de propósito geral em português, utilizando para esse fim uma abordagem baseada em LLMs. De forma mais específica, propõe-se a adaptação da estratégia adotada em Burdisso et al. (2019), Loyola et al. (2021), Loyola et al. (2022) e Thompson et al. (2023), vencedora em diversas tarefas da série de desafios *eRisk* no domínio *R*eddit em inglês, para o caso da detecção precoce de depressão e transtorno de ansiedade em português no domínio do *Twitter*/X brasileiro, usando uma abordagem inédita baseada em LLMs com uso de engenharia de *prompts*.

Nesse cenário, o presente estudo propõe-se a investigar se o uso de uma abordagem baseada em LLM permite maior antecipação de diagnóstico em relação à abordagens tradicionais da área, e se a proximidade das mensagens do momento do diagnóstico na ordem cronológica da *timeline* do *Twitter*/X influencia o resultado do modelo.

As principais contribuições apresentadas nesse estudo são as seguintes:

- Um novo modelo de detecção precoce de depressão/ansiedade em português usando uma abordagem de *prompts* de LLMs, com resultados de antecipação de risco superiores aos de abordagens tradicionais.
- Adaptação do problema de detecção precoce para o caso de redes sociais de propósito geral, como o *Twitter*/X.
- Abordagem de detecção precoce para o caso específico de transtorno de ansiedade, tema que é possivelmente inédito na literatura da área.

O restante desse artigo está organizado da seguinte forma. A seção 2 fundamenta o problema computacional de detecção precoce de transtornos de saúde mental. A seção 3 apresenta um le-

vantamento dos trabalhos correlatos no contexto da série de desafios *eRisk* e iniciativas similares. A seção 4 descreve os modelos propostos nesse trabalho. A seção 5 descreve o córpus utilizado para fins de treino e teste desses modelos, o procedimento de avaliação e seus resultados. Finalmente, a seção 6 sumariza as contribuições apresentadas e destaca possíveis direções de pesquisa futura.

2. Detecção precoce

A detecção precoce (de transtornos de saúde mental ou outros) tal qual proposta nos desafios da série *eRisk* (Losada et al., 2017; Parapar et al., 2023) tem como componente central a métrica de avaliação *f1-latency* introduzida em Sadeque et al. (2018). De acordo com esta definição, para cada usuário $u \in U$ a ser avaliado, todas as suas mensagens são analisadas em ordem cronológica (i.e., conforme aparecem na rede social). Após avaliar k_u ($k_u \geq 1$) mensagens, é tomada então uma decisão binária $d_u \in \{0, 1\}$ atribuindo um rótulo $g_u \in \{0, 1\}$ ao usuário para indicar se é um caso de risco ou não.

Um componente-chave desse tipo de avaliação deve ser o atraso na detecção de verdadeiros positivos, pois se deseja que os sistemas detectem esses casos o quanto antes. Portanto, uma primeira e intuitiva medida de atraso (*latency*) pode ser definida da seguinte forma:

$$\text{latency}_{TP} = \text{median}\{k_u : u \in U, d_u = g_u = 1\}$$

A latência é assim calculada sobre os verdadeiros positivos (TP) detectados pelo sistema, avaliando-se o atraso na decisão com base na mediana de mensagens que o sistema teve que processar para detectar esses casos positivos. Com isso, obtém-se a métrica *f1-latency* proposta em Sadeque et al. (2018), que combina a eficácia da decisão (estimada com a medida *f1* padrão) e o atraso na decisão. O cálculo é realizado multiplicando-se *f1* por um fator de penalidade baseado no atraso médio. Mais especificamente, cada decisão individual (verdadeiro positivo) tomada após a leitura das mensagens k_u recebe a seguinte penalidade:

$$\text{penalty}(k_u) = -1 + \frac{2}{1 + exp^{-p(k_u - 1)}}$$

onde p é um parâmetro que determina a rapidez com que a penalidade deve aumentar. Em Sadeque et al. (2018), p foi definido de forma que a penalidade seja igual a 0,5 do número médio de postagens de um usuário. Nota-se que, para uma decisão logo após a primeira escrita, não há penalidade, ou seja, $penalty(1) = 0$. O fator de velocidade geral do sistema é calculado como:

$$\text{speed} = 1 - \text{median}\{\text{penalty}(k_u) : u \in U, d_u = g_u = 1\}$$

onde a velocidade é igual a 1 para um sistema cujos verdadeiros positivos são detectados logo na primeira escrita. O sistema lento, que (por exemplo) detecta verdadeiros positivos apenas após centenas de mensagens, receberá uma pontuação de velocidade próxima a 0.

Finalmente, a métrica *f1* ponderada pela latência é simplesmente

$$f_{1_{\text{latency}}} = f_1 \cdot \text{speed}$$

Observa-se assim que há uma relação inversa entre a medida *f1* padrão e antecipação, ou seja, quanto mais mensagens forem examinadas, maior tende a ser *f1*, porém menor a possibilidade de antecipação medida por *f1-latency*.

3. Trabalhos relacionados

Variações da medida *f1-latency* discutidas na seção anterior são um dos principais critérios empregados na avaliação dos sistemas participantes da série de desafios (ou *shared tasks*) *eRisk*, que em anos recentes abordaram diversas tarefas computacionais relacionadas a transtornos de saúde mental como depressão, anorexia e outros (Losada et al., 2017, 2018, 2019; Parapar et al., 2022, 2023). Estas tarefas são organizadas com base em um córpus (tipicamente no domínio *Reddit* em inglês) com rotulação binária (risco ou não risco) em nível de usuário. De forma análoga, o mais recente desafio *MentalRiskES* (Mármol-Romero et al., 2023) apresentou pela primeira vez tarefas desse tipo para a língua espanhola. Não há, até onde temos conhecimento, similares voltados ao idioma português.

A Tabela 1 sumariza as tarefas de interesse para o presente trabalho abordadas nesses eventos, com indicativo do tamanho da tarefa em número de instâncias de treino e teste dos córpus utilizados.

3.1. Abordagens selecionadas

Dado que a maioria dos estudos de interesse para a presente pesquisa encontra-se no âmbito da série de desafios *eRisk* dedicado ao idioma inglês, optou-se por realizar um levantamento desses estudos no período 2019-2023, selecionando-se os estudos que obtiveram melhores resultados publicados no contexto destas competições em tarefas

Referência	Idioma	Tarefas	Treino (k)	Teste (k)
Losada et al. (2018)	En	depressão anorexia	531 -	545 -
Losada et al. (2019)	En	anorexia automutilação	254 -	571 170
Losada et al. (2020)	En	automutilação	170	103
Parapar et al. (2021)	En	apostas automutilação	- 170	1129 103
Parapar et al. (2022)	En	Apostas depressão	1128 -	1029 723
Parapar et al. (2023)	En	apostas	1128	1129
Mármol-Romero et al. (2023)	Es	anorexia depressão	6 6	4 5

Tabela 1: Detecção precoce de transtornos de saúde mental em inglês (En) e espanhol (Es).

relacionados à saúde mental. Além desses estudos, foram incluídos também alguns outros trabalhos que não participaram dos desafios *eRisk*, mas que fizeram uso desses conjuntos de dados, conceitos ou métricas de avaliação aplicados à tarefa análoga de detecção precoce de ideação suicida.

A Tabela 2 apresenta um resumo dos estudos selecionados, categorizados conforme o tipo de tarefa abordada (A=anorexia, D=depressão, G=apostas, S=automutilação, SU=ideação suicida), tipo de *feature* textual empregado quando pertinente (emb=*embeddings*, BoW=*bag-of-words*, BoSE=*bag-of-sub-emotions*, DAN=rede de associações profundas, LIWC=*Linguistic Inquiry and Word Count* (Pennebaker et al., 2001), bigrams), e principal método computacional de cada abordagem (SVM=*support vector machine*, SS3=*Smoothness, Significance and Sanction*, MLP=*multilayer perceptron*, ROBERTa, BERT (Devlin et al., 2019), simil.=similaridade textual, LSTM=*long short-term memory networks*, CNN=*convolutional neural networks*, LR=*logistic regression*, RNN=*recurrent neural network*, *Longformer* ou *Ensemble*). Finalmente, é apresentada ainda a posição no *ranking* de resultados da respectiva competição *eRisk*, quando aplicável. Nesta representação, modelos textuais e métodos marcados como '-' indicam que a proposta engloba representação textual e aprendizado em uma solução única (por exemplo, fazendo refinamento de um modelo BERT sem classificação adicional), ou que o trabalho selecionado não foi totalmente claro a esse respeito.

Dado que vários dos sistemas aqui relacionados foram reutilizados em anos subsequentes e em outras tarefas, com ou sem variações significativas em relação à participação inicial, a seguir discutimos apenas os melhores colocados de cada edição da série *eRisk* tomando por base na métrica de avaliação *f1-latency*, detalhada na seção anterior.

O estudo por Mohammadi et al. (2019), que obteve a melhor *f1-latency* da competição *eRisk* em 2019 para a tarefa de detecção de anorexia, propôs uma abordagem comum a diversos outros sistemas participantes da série de desafios. Nessa proposta, é utilizado um conjunto de submodelos neurais baseados em atenção para extrair características e prever probabilidades de classe, posteriormente usadas como entrada para um classificador do tipo SVM.

Na tarefa de detecção de automutilação no mesmo ano, o melhor resultado em termos de *f1-latency* foi obtido pelo estudo em Burdisso et al. (2019). Esse estudo definiu as diretrizes básicas de um modelo que seria reutilizado em tarefas similares nos anos subsequentes da competição, e que tinha como objetivo encontrar um balanço adequado entre a necessidade de maximizar a precisão na identificação de risco e, ao mesmo tempo, minimizar o tempo de uma detecção confiável. Para esse fim, foi proposta uma arquitetura baseada em duas etapas: classificação com informação parcial (*Classification with Partial Information* - CPI) e decisão do momento de classificação (*Deciding the Moment of the Classification* - DMC). Na primeira etapa desta abordagem, somente a classe representando risco é considerada, ou seja, se a entrada parcial for classificada como não-risco, o modelo segue

Domínio	Estudo	Tarefas	Texto	Método	Posição
eRisk2019	Mohammadi et al. (2019)	A	emb.	SVM	1
	Burdisso et al. (2019)	S,A	BoW	SS3	1,5
	Ragheb et al. (2019)	S,A	emb.	MLP	4,2
	Naderi et al. (2019)	S,A	BoW	SVM	2,7
	Aragón et al. (2019)	A	BoSE	SVM	3
	Trifan & Oliveira (2019)	S	BoW	SVM	3,9
	Masood et al. (2019)	A	BoW	SVM	4
eRisk2020	Martínez-Castano et al. (2020)	S	-	ROBERTa	1
	Bagherzadeh et al. (2020)	S	emb.	SVM	3
	Aragón et al. (2020)	S	BoW,BoSE	SVM	5
eRisk2021	Loyola et al. (2021)	G,S	BoW	SVM,SS3	1,1
	Bucur et al. (2021)	G,S	-	BERT	2,3
	Campillo-Ageitos et al. (2021)	S	BoW	SVM	3
	Maupomé et al. (2021)	G,S	emb.	simil.,RoBERTa	4,8
	Lopes (2021)	G,S	emb.	LSTM,CNN	5,11
eRisk2022	Fabregat et al. (2022)	G	emb.	simil.	1
	Srivastava et al. (2022)	G,D	-,BoW	Longformer,SVM	7,1
	Mármol-Romero et al. (2022)	G	-	RoBERTa	2
	Bucur et al. (2022)	G,D	-	RoBERTa	4,2
	Loyola et al. (2022)	G,D	BoW,-	SVM,SS3	3,3
	Sauberli et al. (2022)	D	-	BERT,LR	4
	Ferreira et al. (2022)	G,D	BoW	Ensemble	5,5
eRisk2023	Molina et al. (2023)	G	emb.	SVM	1
	Fabregat et al. (2023)	G	-	RNN	2
	Larrayoz et al. (2023)	G	DAN	FFNN	3
	Thompson et al. (2023)	G	-	BERT,SS3	4
outros	Gollapalli et al. (2021)	SU	emb.	LSTM	-
	Gamoran et al. (2021)	SU	LIWC	LR	-
	Morales et al. (2021)	SU	BoW	Ensemble	-
	Wang et al. (2021)	SU	emb.	SVM	-
	Bayram & Benhiba (2021)	SU	bigrams	Ensemble	-

Tabela 2: Trabalhos relacionados

apenas acumulando mais informações. Na segunda etapa, é aplicada uma política que decide quando parar de ler as entradas e classificar o usuário em questão como sendo de risco caso indicado na etapa anterior. Em versões posteriores desta abordagem, apresentadas em Loyola et al. (2021), Loyola et al. (2022) e Thompson et al. (2023), diversos tipos de representação textual (e.g., *bag-of-words*, *embeddings*, etc.) e métodos de classificação (e.g., SVMs, modelos neurais, etc.) foram experimentados, porém mantendo-se a arquitetura principal baseada nas etapas CPI e DMC. Na competição de 2021, em especial, a abordagem de Loyola et al. (2021) obteve o melhor resultado tanto na tarefa de detecção de vício patológico em apostas quanto na tarefa de detecção de automutilação.

O estudo de Martínez-Castano et al. (2020) obteve o melhor resultado em termos de *f1-latency* para a tarefa de detecção de automutilação do *eRisk* 2020. A abordagem proposta utilizou um modelo do tipo XLM-RoBERTa treinado em um conjunto de textos produzidos por indivíduos da classe positiva (i.e., automutilação), e textos produzidos por indivíduos aleatórios. Dado um conjunto de postagens de teste, a classe a ser atribuída é calculada com base na probabilidade de pertencer a cada um dos dois grupos, utilizando-se de parâmetros ajustados para refletir o limiar de probabilidade média mínima (θ), e as quantidades mínimas e máximas de textos necessários para classificação final (risco ou não risco).

O estudo por Fabregat et al. (2022) utilizou uma abordagem de vizinhos mais próximos para detecção precoce de vício em apostas, obtendo o melhor resultado de *f1-latency* para esta tarefa no contexto do desafio *eRisk* 2022. Nesta abordagem, uma mensagem é classificada como risco se os 20 vizinhos mais próximos também correspondem a mensagens consideradas de risco. Além disso, foi adotada uma estratégia de recriação dos rótulos do córpus de treinamento de modo a gerar uma rotulagem em nível de mensagem (e não em nível de usuário, que é o padrão nas tarefas *eRisk* aqui discutidas). Para esse fim, todas as mensagens de entrada inicialmente recebem o mesmo rótulo do seu autor (usuário), e são comparadas a subconjuntos de mensagens semelhantes, recebendo o rótulo do conjunto mais próximo. Nesse processo, assume-se que usuários da casse positiva podem conter mensagens negativas mas não o contrário, pois se usuários da classe negativa pudessem apresentar mensagens de teor positivo, então seriam rotulados como positivos.

No caso da tarefa de detecção precoce de depressão da mesma competição, o melhor resultado de *f1-latency* foi apresentado pelo estudo em Srivastava et al. (2022). Nesta abordagem, utilizaram-se diferentes métodos de engenharia de características e técnicas de classificação de texto. Na tarefa de predição de depressão, optou-se pelo uso de um modelo do tipo *bag-of-words* com ponderação tf-idf baseada na medida de entropia, em conjunto com um classificador SVM.

Finalmente, o estudo em Molina et al. (2023) também utilizou uma abordagem tradicional de aprendizado de máquina baseada em SVM com um modelo de contagens tf-idf de unigramas. Esta abordagem foi aplicada à detecção de vício patológico em apostas na competição *eRisk* 2023, obtendo o melhor resultado para a tarefa.

3.2. Considerações

Com base no levantamento realizado, observa-se que os estudos mais bem-sucedidos da área na série de desafios *eRisk* tendem a apresentar soluções computacionais relativamente simples se comparadas ao estado da arte de outras tarefas do PLN. Embora métodos mais atuais tenham sido recorrentes em todas as edições da série, as abordagens tradicionais apresentaram, de modo geral, resultados mais robustos. Uma possível explicação para esse cenário pode estar na própria natureza de eventos desse tipo, que tendem a privilegiar a entrega rápida de resultados com menor margem para a exploração de métodos de custo computacional elevado. Assim, métodos mais contemporâneos, como o uso de grandes modelos de língua (LLMs) como pretendido no presente trabalho, permanecem em grande parte abertos à investigação.

De especial interesse para o presente trabalho, são dignos de nota os resultados expressivos da arquitetura de duas etapas aplicada a diversas tarefas em Burdisso et al. (2019), Loyola et al. (2021), Loyola et al. (2022) e Thompson et al. (2023). Esta arquitetura será adaptada às tarefas de detecção de depressão e transtorno de ansiedade representadas pelo córpus *SetembroBR*, e utilizada tanto como ponto de partida para a presente estratégia – baseada em LLMs – como para a definição de modelos de *baseline*.

4. Detecção precoce de depressão/ansiedade em português

Esta seção aborda os problemas da detecção precoce de depressão e transtorno de ansiedade representados no córpus SetembroBR (dos Santos et al., 2023) com uso de LLMs, e a questão de como esta formulação do problema (baseado na rede social *Twitter*/X em português) se compara ao originalmente considerado no domínio *Reddit* na série de desafios *eRisk* (Losada et al., 2017) em inglês. Assim, o presente estudo propõe-se a investigar as seguintes questões de pesquisa:

Q1 O uso de uma abordagem baseada em LLM para detecção precoce de risco de transtorno de saúde mental permite maior antecipação de diagnóstico em relação à abordagens tradicionais da área?

Q2 A proximidade das mensagens do momento do diagnóstico na ordem cronológica da *timeline* do *Twitter*/X influencia a tarefa de detecção precoce de transtornos de saúde mental?

A questão de pesquisa $Q1$ é motivada pelo interesse em empregar métodos computacionais mais recentes à presente tarefa — baseadas no uso de LLMs — tendo em vista uma possível melhoria nos resultados. Entretanto, mesmo sendo esperado algum incremento na acurácia da classificação textual com o uso de um modelo mais sofisticado desse tipo, o principal interesse desta investigação, assim como na série de desafios *eRisk* (Parapar et al., 2023), está na questão da antecipação do diagnóstico, ou seja, na capacidade do modelo realizar a classificação com base no menor volume de dados (i.e., postagens) possível.

A questão $Q2$ é motivada pela observação de que, ao contrário dos grupos de discussão sobre saúde mental da plataforma *Reddit* empre-

gados na série de desafios *eRisk*, a rede social *Twitter*/X possui mensagens tratando dos mais diversos assuntos, e não apenas transtornos de saúde mental. Além disso, como estas mensagens são normalmente publicadas em longos períodos de tempo, é possível que boa parte dos dados disponíveis tenha pouca ou nenhuma relevância para a saúde mental. Considerando-se ainda que estudos como Eichstaedt et al. (2018) sugerem que a porção de dados relevantes para a tarefa estaria concentrada dentro dos 3 meses de atividade mais recente na rede social, o foco desta segunda investigação é o efeito da ordenação temporal de mensagens – ou proximidade do momento do diagnóstico – presente no *Twitter*/X, porém ausente no domínio da série de desafios *eRisk* (Losada et al., 2017, 2018, 2019, 2020; Parapar et al., 2021, 2022, 2023).

4.1. Modelos desenvolvidos

Para investigação das questões de pesquisa $Q1$ e $Q2$ introduzidas na seção anterior, foi desenvolvida uma abordagem baseada em predições obtidas com auxílio do modelo GPT 3.5[1], aqui denominado *GPTrelev*, e duas alternativas de *baseline* denominadas *W2V.logreg* e *Soft.BERT* que são adaptações para o domínio do *Twitter*/X em Português do modelo vencedor da série de desafios *eRisk* 2022 (Parapar et al., 2022) apresentado em Loyola et al. (2022). O código da implementação desses modelos está publicamente disponível[2].

Tanto o modelo *GPTrelev* proposto como os *baselines* *W2V.logreg* e *Soft.BERT* seguem a mesma arquitetura geral proposta de Loyola et al. (2022). Esta arquitetura conta com dois módulos principais: um módulo de classificação parcial (CPI), e outro de decisão do momento da classificação (DMC) responsáveis, respectivamente, pela classificação em nível de postagens e pela consolidação destas classificações em uma predição final em nível de usuário (ou *timeline*).

O módulo CPI realiza a leitura e classificação dos dados de entrada em lotes de m mensagens (*tweets*), atribuindo um rótulo binário (diagnosticado ou controle) a cada grupo. Com base nos grupos de mensagens assim classificadas, o módulo DMC procura então estimar, com a maior antecipação possível, se o indivíduo que as publicou possui ou não risco de vir a receber um diagnóstico de depressão/ansiedade. Assim, a decisão é tomada com base no menor volume possível de dados classificados pelo primeiro módulo.

[1] https://platform.openai.com/docs/models

[2] https://github.com/brunoissamo/mestrado_deteccao_precoce_redes_sociais

O módulo DMC utiliza uma adaptação da política de decisão proposta de Loyola et al. (2022). Conforme descrito na seção 3, o módulo DMC determina, a partir do resultado dado pelo módulo CPI, qual o momento adequado para interromper a leitura de *tweets* e de fato realizar a classificação do indivíduo como sendo do grupo de risco ou não. Se a probabilidade de diagnóstico for superior a um δ pré-definido, a leitura é interrompida e o indivíduo é classificado como tendo risco acima da população em geral de vir a obter um diagnóstico de depressão/ansiedade. Caso contrário, o sistema continua a leitura do próximo lote de m mensagens até o fim da *timeline*, podendo eventualmente ser classificado como pertencente ao grupo controle, ou seja, sem risco acima da média da população geral de vir a receber um diagnóstico desse tipo. Os detalhes específicos da adaptação desta política nos modelos propostos são discutidos nas seções a seguir.

4.1.1. Modelo GPTrelev

O modelo *GPTrelev* proposto combina a política de análise de risco proposta de Loyola et al. (2022) com predições obtidas com auxílio do modelo GPT 3.5. A escolha desse LLM foi motivada pelos resultados positivos observados em tarefas análogas de predição de estados de saúde mental com base no córpus *SetembroBR* como as discutidas em dos Santos & Paraboni (2024), dentre outras.

Em linhas gerais, o modelo *GPTrelev* substitui o módulo de análise das publicações de Loyola et al. (2022), originalmente composto de um classificador SVM e *word embeddings* computados com uso de *word2vec* (Mikolov et al., 2013), por predições obtidas diretamente do LLM. A arquitetura geral do modelo *GPTrelev* é ilustrada na Figura 1.

Conforme ilustrado na Figura 1, a principal diferença entre o modelo *GPTrelev* e a proposta em Loyola et al. (2022) é a substituição das probabilidades dadas pelo classificador SVM por escores indicativos de transtorno de saúde mental em uma escala de 0 a 10, em que 0 indica baixos indícios de risco, que no presente trabalho foram obtidas a partir do estudo em dos Santos & Paraboni (2024) . De forma mais específica, foi submetido ao modelo GPT um *prompt* solicitando que cada mensagem fosse analisada buscando indícios linguísticos que pudessem sugerir a presença de sintomas contínuos de depressão, atribuindo um escore de 0 a 10 que pode ser visto como grau de afinidade de cada mensagem com o

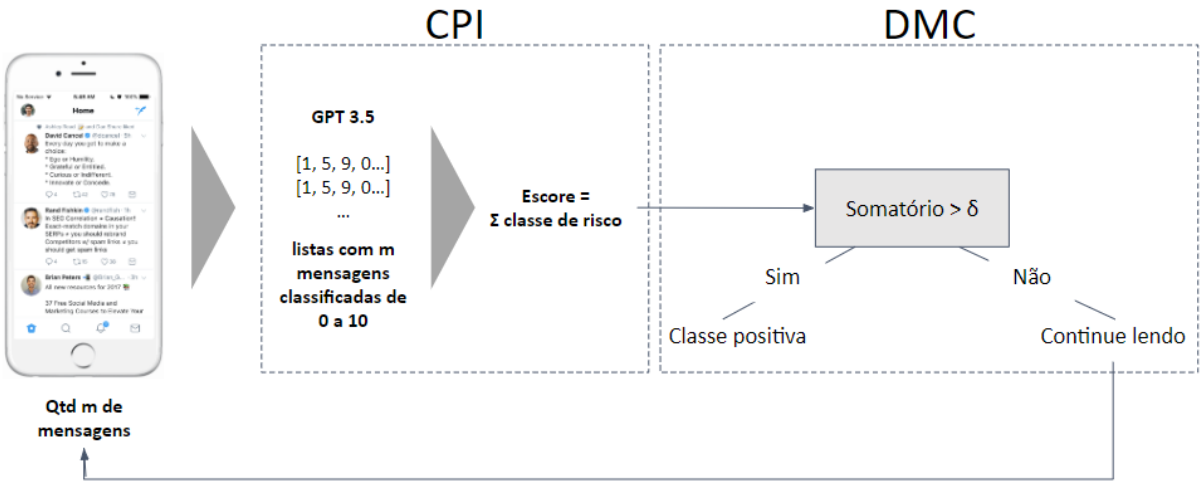


Figura 1: Arquitetura *GPTrelev*

tema de saúde mental, considerando expressões persistentes de desespero, tristeza profunda ou ansiedade constante como marcadores relevantes para realizar as classificações. A Tabela 3 sumariza a distribuição de escores obtidos sobre os conjuntos de teste do córpus *SetembroBR* para as tarefas de detecção de depressão e ansiedade.

	Depressão		Ansiedade	
Relevância	***Tweets***	%	***Tweets***	%
0	40.561	18,1	54.314	17,2
1	50.928	22,8	72.219	22,9
2	81.318	36,4	117.191	37,1
3	28.416	12,7	40.488	12,8
4	567	0,3	770	0,2
5	3	0,0	8	0,0
6	20.442	9,1	29.055	9,2
7	1.292	0,6	1.866	0,6
8	182	0,1	217	0,1
9	0	0,0	0	0,0
10	2	0,0	0	0,0

Tabela 3: Distribuição de escores de relevância para saúde mental produzidos pelo modelo GPT em dos Santos & Paraboni (2024)

Conforme observado na Tabela 3, o LLM de modo geral evita as categorias mais centrais, quase sempre comprometendo-se com uma resposta de baixa ou alta relevância. Além disso, observa-se que as categorias superiores (a partir de 8, e principalmente 9 e 10) praticamente nunca são selecionadas. O mesmo comportamento geral é observado no conjunto dos dados de ansiedade.

Na arquitetura *GPTrelev*, os escores de relevância gerados pelo LLM são somados em janelas de m mensagens, e esse resultado é fornecido ao módulo DMC (conforme Figura 1) para tomada de decisão. Caso esta soma seja maior que um valor δ, o usuário é classificado como diagnosticado. Do contrário, o sistema continua a leitura da próxima janela de mensagens. Foram testados diversos valores para os parâmetros δ e m, escolhendo aqueles que obtivessem maior valor da métrica *f1-latency* em um conjunto de validação do córpus. Para m foram testados os valores ímpares de 5 a 19 e para δ os valores $1, 5$, 2, $2,5$ e 3, sendo ao final utilizados $m = 15$ e $\delta = 2,5$. Cabe ressaltar que foi testado também o uso de janelas acumulativas dos dados e os resultados não foram expressivos, por isso optou-se por processar uma quantidade menor de informação, com janela móveis, e obter melhores métricas de desempenho.

4.1.2. Modelos de baseline

Como sistemas de *baseline* para comparação com o modelo GPTrelev descrito na seção anterior, foram consideradas duas abordagens baseadas na proposta original em Loyola et al. (2022), adaptadas ao domínio da rede social *Twitter*/X em português. Estas adaptações, que consistiram em encontrar novos valores para os parâmetros m e δ da abordagem original e usar *embeddings* para o português, foram necessárias em razão de diferenças na definição do problema de classificação (diagnosticados versus controle no córpus *SetembroBR* e depressivos versus não depressivos em Parapar et al. (2023)), gênero linguístico (rede social de propósito geral versus fórum de discussão sobre saúde mental), tamanho das publicações (*tweets* curtos versus postagens *Reddit* longas) e idioma (português versus inglês).

Assim como no modelo *GPTrelev* proposto, os dois modelos de *baseline* utilizados mantêm fixa a política de decisão, variando-se apenas a forma como as publicações individuais são avali-

adas. A arquitetura desses dois modelos é ilustrada na Figura 2 e detalhada a seguir.

Conforme ilustrado na Figura 2, para cada janela com m mensagens (*tweets*) o classificador do módulo CPI produz rótulos 0 (não diagnosticado) ou 1 (diagnosticado) utilizados na primeira etapa do DMC. Caso esse rótulo seja 0, o sistema continua lendo a próxima janela. Do contrário, verifica se a probabilidade obtida no primeiro módulo é maior que um valor pré-definido δ. Caso a probabilidade seja maior, o sistema interrompe a leitura e rotula o usuário como sendo diagnosticado. Caso contrário, continua lendo as mensagens até que finalize a *timeline*. O valor do parâmetro δ foi o mesmo utilizado no trabalho original em Loyola et al. (2022), como sendo 0,5.

Do ponto de vista da implementação, o modelo *W2V.logreg* faz uso de um classificador do tipo regressão logística e um modelo textual do tipo *bag-of-words*. Esta estratégia, embora não tenha sido a melhor combinação da competição *eRisk* 2022 (que usa um classificador do tipo SVM) apresentou os melhores resultados em estudos preliminares com o córpus *SetembroBR*.

Em complemento ao *baseline* *W2V.logreg*, considerou-se uma segunda alternativa, o modelo *Soft.BERT*, o qual também foi inspirado no modelo desenvolvido em Loyola et al. (2022), porém nesse caso substituindo-se a abordagem BoW com SVM por um classificador baseado no modelo BERT com saída *softmax*. Para adaptação ao contexto da língua portuguesa, realizou-se o *fine-tuning* na arquitetura bertabaporu-base-uncased (da Costa et al., 2023), um modelo BERT treinado em postagens do Twitter/X brasileiro como parte do projeto *SetembroBR*, e que inclui postagens produzidas pelos próprios usuários integrantes do córpus. Dado que em testes preliminares o uso de um número maior de épocas não revelou ganho significativo, por razões de custo computacional optou-se por realizar o *fine-tuning* com uso do otimizador AdamW por 3 épocas com taxa de aprendizado de 2e-5. Assim como no caso anterior, dada uma sequência de *tweets* de entrada obtém-se como resultado uma classificação e uma probabilidade, que será utilizado no módulo DMC, conforme ilustrado na Figura 2.

5. Avaliação

Como forma de investigar as questões de pesquisa $Q1$ e $Q2$ introduzidas na seção anterior, foram conduzidos dois experimentos com uso dos modelos *GPTrelev*, *W2V.logreg* e *Soft.BERT* descritos na seção 4.1. Nas seções a seguir é apresentado o conjunto de dados e procedimento de treino desses modelos, que é comum a ambos experimentos, e os resultados individuais de cada questão de pesquisa, seguidos de uma breve análise de erros.

5.1. Conjunto de dados

O presente trabalho é baseado no córpus *SetembroBR* apresentado em sua forma final em dos Santos et al. (2023) a partir do estudo-piloto em dos Santos et al. (2020). O córpus consiste de uma coleção de postagens (ou *tweets*) da rede social *Twitter*/X do Brasil de indivíduos diagnosticados com algum tipo de transtorno de depressão ou ansiedade, denominada classe *Diagnosticados*, e de uma classe *Controle* formado por indivíduos selecionados aleatoriamente, e que é de proporção sete vezes superior à classe *Diagnosticados*.

As *timelines* das classes *Diagnosticados* e *Controle* são pareadas por gênero (masculino e feminino), quantidade de publicações e período de publicação. Assim, para cada indivíduo da classe *Diagnosticado* há sete indivíduos da classe *Controle* com estas mesmas características. Nesta formulação do problema, amplamente adotada em diversos estudos da área (Coppersmith et al., 2015; Losada et al., 2017; Lynn et al., 2018; Cohan et al., 2018; Losada et al., 2019; Parapar et al., 2022), a tarefa computacional a ser modelada consiste em identificar indivíduos com probabilidade acima da média da população em geral (representada pela classe *Controle*) de vir a receber um futuro diagnóstico de depressão/ansiedade, e não se trata portanto de distinguir, por exemplo, indivíduos depressivos de não-depressivos.

No caso dos indivíduos diagnosticados, as publicações integrantes do córpus *SetembroBR* são limitadas ao período anterior ao momento em que o diagnóstico foi emitido por um profissional da área de saúde. O córpus é assim especialmente voltado ao desenvolvimento de aplicações de predição de depressão/ansiedade a partir do histórico de publicações que antecede o reconhecimento do transtorno e eventual tratamento.

A Tabela 4, adaptada de dos Santos et al. (2023), apresenta estatísticas descritivas da porção textual do córpus *SetembroBR* a ser utilizada no presente trabalho.

O córpus *SetembroBR* possui uma divisão aleatória padrão entre *timelines* de treinamento (80%) e teste (20%). A divisão existente será adotada também nos experimentos aqui relatados.

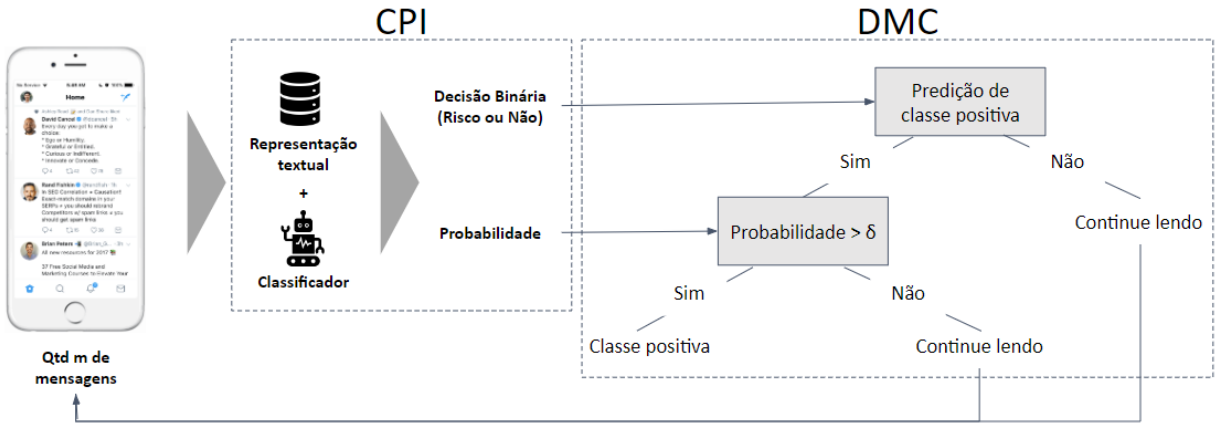


Figura 2: Descrição da Política de Decisão (PD) - *baselines*

	Depressão			Ansiedade		
Métrica	Diagnost.	Controle	Total	Diagnost.	Controle	Total
Usuários	1.684	11.788	13.472	2.219	15.533	17.752
Tweets (milhões)	2,43	16,99	19,42	3,43	23,98	27,41
Tokens (milhões)	29,32	201,94	231,26	42,24	281,51	323,75
Tweets/usuários	1.441	1.441	1.441	1.543	1.543	1.543
Tokens/tweets	12,08	11,88	11,91	12,33	11,74	11,81

Tabela 4: Estatísticas descritivas do córpus SetembroBR (dos Santos et al., 2023)

5.2. Procedimento

O treinamento e teste dos modelos *GPTrelev*, *W2V.logreg* e *Soft.BERT* fez uso das respectivas porções padrão de treino/teste do córpus *SetembroBR*. No caso do teste, entretanto, são usadas porções de dados diferentes para investigar as questões de pesquisa $Q1$ e $Q2$. Para a questão $Q1$, foram utilizadas apenas r mensagens mais recentes de cada *timelines* do conjunto de teste, sendo r o tamanho da menor *timeline* do córpus. Esta decisão é motivada pelo estudo em Eichstaedt et al. (2018), que sugere que os dados relevantes para fins de detecção de depressão concentram-se nos três meses mais recentes do histórico de mensagens de um indivíduo, e também por razões de custo computacional. Para a questão $Q2$, por outro lado, é utilizada a *timeline* completa de cada usuário.

O treinamento do modelo *GPTrelev* utilizou os rótulos de relevância para saúde mental produzidos pelo LLM para obter os parâmetros de decisão m e δ. Também por razões de custo, e dado que o córpus é de um gênero textual único e com certa sobreposição de dados (parte dos indivíduos diagnosticados com um transtorno é também diagnosticado com outro, e a classe Controle é parcialmente a mesma nos dois conjuntos), esses parâmetros foram posteriormente usados tanto para o classificador de depressão como ansiedade.

Seguindo Parapar et al. (2023), a avaliação foi realizada computando-se a medida *f1-latency*, que representa a ponderação entre o acerto e a quantidade de mensagens lidas. Além disso, dado que o córpus *SetembroBR* disponibiliza informação de data e hora das publicações, como forma de ilustrar o eventual ganho (i.e., antecipação de diagnóstico) obtido pelos modelos avaliados de forma possivelmente mais intuitiva, foi computada também a média de dias em que um diagnóstico poderia ser antecipado. Esse ganho é calculado como a média de dias transcorridos entre a data do último *tweet* da *timeline* e do último lido pelo sistema para cada usuário, dado em quantidade de dias. Finalmente computou-se também a medida *f1* da classe positiva como forma de destacar a relação inversa entre esta medida e *f1-latency*, acompanhada da medida *f1-macro* tradicional, a qual apresenta uma visão menos clara do desempenho do modelo na classe de interesse (ou seja, dos indivíduos diagnosticados) em virtude do maior tamanho da classe negativa (ou grupo Controle aleatório).

5.3. Experimento 1: o uso de LLM para detecção precoce de depressão/ansiedade

A Tabela 5 sumariza os resultados obtidos pelos modelos *GPTrelev*, *W2V.logreg* e *Soft.BERT* com

base na porção de teste do córpus *SetembroBR* no contexto da questão de pesquisa $Q1$, que trata do uso da abordagem baseada em LLM – o modelo *GPTrelev* – na detecção precoce de risco de depressão e transtorno de ansiedade.

Com relação à tarefa de detecção de depressão (na parte superior da Tabela 5), os resultados indicam que, de modo geral, o modelo *GPTrelev* é superior às abordagens de *baseline* de acordo com as métricas consideradas. Esta vantagem é evidente tanto no que diz respeito à antecipação do diagnóstico (representada pela medida *f1-latency*) como no que diz respeito à acurácia de classificação de diagnósticos (representada pela medida *f1-pos* da classe positiva). A exceção é observada no resultado de *f1-macro* do modelo *W2V.logreg*, que foi ainda ligeiramente superior ao resultado do modelo *GPTrelev*.

Com relação à tarefa de detecção de transtorno de ansiedade (parte inferior da Tabela 5), os resultados indicam novamente uma superioridade do modelo *GPTrelev*, embora o número de dias de antecipação do diagnóstico do modelo *Soft.BERT* tenha sido ligeiramente superior.

Considerando-se a questão da antecipação de risco representada pelas métricas *f1-latency* e número médio de dias de antecipação, observa-se tanto na detecção precoce de depressão como de transtorno de ansiedade uma divisão clara entre os modelos avaliados. Em um extremo, observa-se que os resultados de antecipação obtidos pelos modelos *Soft.BERT* e *GPTrelev* são relativamente próximos, enquanto que os do modelo *W2V.logreg* ocupam um lugar mais distante mesmo considerando-se que esta diferença não tenha tido grande reflexo em perda de acurácia (representada pelas medidas *f1-pos* e *f1-macro*). Assim, a adaptação do modelo *W2V.logreg*, que procura ser fiel à proposta original em Loyola et al. (2022), parece não ter sido tão bem sucedida no domínio *SetembroBR* quanto as alternativas aqui consideradas. Este resultado pode ter sido influenciado tanto pelas diferenças de gênero textual, de idioma e de definição da tarefa entre os córpus *eRisk* e *SetembroBR*, mas mais estudos são necessários para determinar porque estas diferenças são mais pronunciadas no uso de *embeddings* estáticos do tipo *word2vec*, e menor nas alternativas baseadas em BERT e GPT.

De forma conjunta, esses resultados oferecem uma resposta parcial à questão de pesquisa $Q1$, sugerindo que o uso de LLMs na forma proposta para detecção precoce de risco de transtorno de saúde mental permite maior antecipação de diagnóstico em relação a abordagens tradicionais desta área.

5.4. Experimento 2: o papel da informação temporal na predição precoce de depressão/ansiedade

Apesar da relativa superioridade do modelo baseado em LLM discutidos na seção anterior, os resultados obtidos para a tarefa de detecção precoce no córpus *SetembroBR* são claramente inferiores aos observados em tarefas análogas como as desenvolvidas pelos participantes dos desafios *eRisk* (Losada et al., 2017). Conforme discutido na seção anterior, esta diferença pode estar relacionada ao fato de que a natureza do problema – e dos dados – é distinta em pelo menos dois aspectos. Em primeiro lugar, há uma diferença de conteúdo a ser considerada. Ao contrário de grupos de discussão sobre saúde mental no *Reddit*, o córpus *SetembroBR* contém *timelines* tratando de qualquer tema. Em segundo lugar, há a noção de proximidade em relação ao momento do diagnóstico na ordem cronológica das *timelines* do *Twitter*/X, especialmente considerando-se que somente dados mais recentes são tidos como relevantes para esse fim em estudos como Eichstaedt et al. (2018).

Como forma de investigar estas diferenças, foi realizada uma análise comparativa dos resultados já apresentados na seção anterior, referentes ao modelo *GPTrelev* original (que recebe como entrada a porção da *timeline* de cada indivíduo de teste mais próxima possível do momento do diagnóstico), com resultados de uma versão modificada desse mesmo modelo, que recebe como entrada a *timeline* completa de cada indivíduo. O objetivo desta comparação foi o de verificar se, em uma situação prática em que o momento do diagnóstico não é conhecido (ou seja, tomando-se como entrada um conjunto de dados qualquer, que pode ou não conter mensagens de maior relevância para a detecção do transtorno) haveria perda de desempenho.

A Tabela 6 apresenta os resultados do modelo *GPTrelev* utilizando as postagens mais recentes (reproduzidos do experimento anterior para fins ilustrativos) e os resultados do mesmo modelo com base na *timeline* completa.

Os resultados apresentados na Tabela 6 mostram que, em linhas gerais, realizar a leitura sequencial de mensagens considerando-se apenas as mais próximas do momento do diagnóstico – como originalmente proposto no experimento da seção anterior – é realmente superior à leitura a partir da primeira mensagem disponível na *timeline*. Assim, ao contrário dos grupos de discussão *Reddit* da série *eRisk* (Parapar et al., 2023), a detecção precoce com base em *timelines* do *Twitter*/X é dependente do momento da

Tarefa	Modelo	f1-latency	Antecip. (dias)	f1-pos	f1-macro
Depressão	W2V.logreg	0,20	77	0,25	**0,59**
	Soft.BERT	0,20	**453**	0,22	0,14
	GPTrelev	**0,24**	436	**0,28**	0,56
Ansiedade	W2V.logreg	0,15	47	0,15	0,54
	Soft.BERT	0,21	**508**	0,22	0,47
	GPTrelev	**0,23**	479	**0,27**	**0,55**

Tabela 5: Q1: abordagem baseada em LLM GPTrelev e abordagens tradicionais. O melhor desempenho de acordo com cada métrica de avaliação é destacado.

Tarefa	Timeline	f1-latency	Antecip. (dias)	f1-pos	f1-macro
Depressão	Completa	0,13	296	0,27	0,39
	Mais próxima	**0,24**	**436**	**0,28**	**0,56**
Ansiedade	Completa	0,11	346	0,25	0,37
	Mais próxima	**0,23**	**479**	**0,27**	**0,55**

Tabela 6: Resultados do modelo GPTrelev com timeline completa e mais próxima do diagnóstico. O melhor desempenho de acordo com cada métrica de avaliação é destacado.

amostragem, o que responde de forma afirmativa à presente questão de pesquisa *Q*2. De forma mais específica, é provável que, em uma rede social de postagens ordenadas cronologicamente como no caso do *Twitter*/X, uma abordagem de detecção precoce apresentaria resultados ótimos apenas se os limites de informação temporal tipicamente considerados para fins de diagnóstico forem respeitados, como por exemplo o limite de três meses proposto em Eichstaedt et al. (2018). Este resultado é também consistente com as diretrizes de diagnóstico de depressão proposta na área de psicologia (American Psychiatric Association, 2013), que privilegiam o caráter recorrente e persistente de sintomas, e não sua ocorrência em episódios isolados.

5.5. Análise de erros

Como forma de identificar possíveis fontes de erro na abordagem *GPTrevel* proposta, foi realizada uma análise da distribuição das mensagens de diferentes graus de relevância para a tarefa utilizadas na presente abordagem, e que foram geradas com uso de um modelo GPT em dos Santos & Paraboni (2024). De forma mais específica, foram contabilizadas as quantidades de mensagens de grau de relevância baixa (escores de relevância 0, 1 e 2), média (escores 3, 4 e 5) e alta (6, 7 e 8) encontradas nas instâncias classificadas correta e incorretamente de cada classe (Diagnosticados ou Controle). Conforme discutido na seção 4.1, entretanto, é importante ressaltar que as mensagens de baixa relevância não fazem parte dos dados utilizados pelo modelo *GPTrevel*, sendo incluídas nesta análise apenas para fins comparativos.

A Tabela 7 apresenta os resultados obtidos para as tarefas de detecção de depressão e transtorno de ansiedade, com indicativo dos casos em que houve aumento (↑) ou diminuição (↓) do percentual de mensagens nas situações de classificação incorreta.

Tanto na tarefa de detecção de depressão como ansiedade, os resultados da Tabela 7 sugerem efeitos semelhantes. Nos casos incorretos da classe Diagnosticados com depressão/ansiedade, observa-se que as mensagens de média e alta relevância são *menos* frequentes. Inversamente, nos casos incorretos da classe Controle para depressão/ansiedade, as mensagens de média e alta relevância são *mais* frequentes.

Estes resultados indicam que a rotulação de relevância é uma possível fonte de erros de classificação tanto para a classe positiva como negativa, embora não seja possível afirmar a causa exata destas divergências. Uma possível explicação estaria no próprio projeto do córpus *SetembroBR*, cuja classe Controle naturalmente oculta um certo número de usuários aleatórios que podem ter depressão/ansiedade não relatada (dos Santos et al., 2023). Outra possibilidade, entretanto, é que a própria rotulação atribuída pelo modelo GPT em dos Santos & Paraboni (2024) apresente imprecisões.

	Depressão				Ansiedade			
	Diagnosticados		Controle		Diagnosticados		Controle	
Relev.	Corretos	Incorretos	Corretos	Incorretos	Corretos	Incorretos	Corretos	Incorretos
baixa	94,4%	95,33% ↑	98,68%	97,9% ↓	62,92%	65,76% ↑	81,45%	77,78% ↓
média	2,88%	2,67% ↓	0,98%	1,49% ↑	17,53%	17,32% ↓	11,42%	13,31% ↑
alta	2,71%	1,99% ↓	0,35%	0,62% ↑	19,56%	16,92% ↓	7,13%	8,90% ↑

Tabela 7: Distribuição de postagens de diferentes graus de relevância nas instâncias classificadas correta e incorretamente na detecção de depressão e ansiedade.

6. Considerações finais

Este artigo apresentou uma abordagem para detecção precoce de transtornos de saúde mental a partir de publicações em redes sociais baseada em LLMs, a qual foi comparada a abordagens tradicionais da área. Além disso, levando-se em conta a natureza das publicações em uma rede social de propósito geral, e sua distribuição ao longo de uma *timeline* de publicações, foi investigada também a adaptação do problema tal qual formulado na série de desafios *eRisk* (Parapar et al., 2023) – originalmente baseado em fóruns de discussão *Reddit* – para o domínio do *Twitter*/X brasileiro.

No que diz respeito à primeira questão – sobre o uso de LLMs na tarefa – os resultados obtidos sugerem que LLMs permitem maior antecipação do diagnóstico em relação à abordagem tradicionais, e que o uso de mensagens mais recentes demonstrou-se mais útil na detecção precoce de depressão e ansiedade.

Quanto à adaptação do problema de detecção precoce à rede social *Twitter*/X, observa-se que que há necessidade de filtrar mensagens de baixa relevância para a tarefa (e que são normalmente a maioria das mensagens em uma rede social de propósito geral), e levar em conta as publicações realizadas em momento mais próximo ao agravamento do problema (ou seus sintomas, etc.) já que, por exemplo, mesmo mensagens potencialmente relevantes (e.g., problemas como insônia) não são necessariamente indicativos de depressão se ocorrem de forma isolada ou distante do momento analisado (American Psychiatric Association, 2013).

Este estudo deixa uma série de oportunidades de melhoria. Em primeiro lugar, observa-se que os resultados, principalmente no que diz respeito à acurácia de classificação, são inferiores ao observados no domínio *eRisk*. Isso sugere que a adaptação ao domínio *Twitter*/X é uma tarefa complexa, e ainda não totalmente resolvida no presente trabalho. Uma possível explicação para esta diferença, e que carece de mais estudos, pode ser o fato de que, em uma rede social como o *Twitter*/X, as mensagens ditas relevantes para a tarefa são concentradas em pontos específicos da *timeline* (i.e., próximas ao momento em que o transtorno se agrava ou manifesta sintomas), o que possui impacto considerável sobre a capacidade do modelo de antecipar risco em uma leitura sequencial de mensagens ordenadas cronologicamente. Em outras palavras, dificuldades desse tipo podem não ser observadas, ou podem ser observadas em menor escala, na tarefa de detecção precoce baseada em fóruns de discussão especializada em saúde mental.

Finalmente, a própria questão do quê constitui uma mensagem relevante para a tarefa de detecção precoce de transtornos de saúde mental permanece em aberto. No presente trabalho, foi utilizado um método baseado em *prompts* submetidos ao modelo GPT para esta finalidade. Não está claro, entretanto, se esse método é realmente ideal para esta tarefa, ou como ele se compara a alternativas similares que seriam possíveis, por exemplo, utilizando outros LLMs. A investigação destas questões é também deixada como sugestão de trabalhos futuros.

Agradecimentos

Esse trabalho contou com apoio FAPESP #2021/08213-0.

Referências

Agrawal, Aryan. 2024. Illuminate: A novel approach for depression detection with explainable analysis and proactive therapy using prompt engineering. ArXiv [cs.CL]. 10.48550/arXiv.2402.05127

Amanat, Amna, Muhammad Rizwan, Abdul Rehman Javed, Maha Abdelhaq, Raed Alsaqour, Sharnil Pandya & Mueen Uddin. 2022. Deep learning for depression detection from textual data. *Electronics* 11(5). 676. 10.3390/electronics11050676

American Psychiatric Association. 2013. *Diagnostic and statistical manual of mental disorders*. American Psychiatric Association 5th edn. 10.1176/appi.books.9780890425596

Aragón, Mario Ezra, Adrián Pastor López-Monroy & Manuel Montes-y Gómez. 2019. INAOE-CIMAT at eRisk 2019: Detecting signs of anorexia using fine-grained emotions. Em *CLEF 2019 Working Notes*,

Aragón, Mario Ezra, Adrián Pastor López-Monroy & Manuel Montes-y Gómez. 2020. INAOE-CIMAT at eRisk 2020: Detecting signs of self-harm using sub-emotions and words. Em *CLEF 2020 Working Notes*,

Bagherzadeh, Hojjat, Ehsan Fazl-Ersi & Abedin Vahedian. 2020. Detection of early sign of self-harm on Reddit using multi-level machine. Em *CLEF 2020 Working Notes*,

Bayram, Ulya & Lamia Benhiba. 2021. Determining a person's suicide risk by voting on the short-term history of Tweets for the CLPsych 2021 shared task. Em *7th Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 81–86. 10.18653/v1/2021.clpsych-1.8

Bevendorff, Janek, Berta Chulvi, Elisabetta Fersini, Annina Heini, Mike Kestemont, Krzysztof Kredens, Maximilian Mayerl, Reynier Ortega-Bueno, Piotr Pezik, Martin Potthast, Francisco Rangel, Paolo Rosso, Efstathios Stamatatos, Benno Stein, Matti Wiegmann, Magdalena Wolska & Eva Zangerle. 2022. Overview of PAN 2022: Authorship verification, profiling irony, stereotype spreaders, and style change detection. Em *Advances in Information Retrieval*, 382–394. 10.1007/978-3-030-99739-7_42

Bucur, Ana-Maria, Adrian Cosma & Liviu P Dinu. 2021. Early risk detection of pathological gambling, self-harm and depression using BERT. Em *CLEF 2021 Working Notes*, 928–937.

Bucur, Ana-Maria, Adrian Cosma, Liviu P Dinu & Paolo Rosso. 2022. An end-to-end set transformer for user-level classification of depression and gambling disorder. Em *CLEF 2022 Working Notes*, 851–863.

Burdisso, Sergio G, Marcelo Errecalde & Manuel Montes-y Gómez. 2019. UNSL at eRisk 2019: a unified approach for anorexia, self-harm and depression detection in social media. Em *CLEF 2019 Working Notes*,

Campillo-Ageitos, Elena, Hermenegildo Fabregat, Lourdes Araujo & Juan Martinez-Romo. 2021. NLP-UNED at eRisk 2021: self-harm early risk detection with TF-IDF and linguistic features. Em *CLEF 2021 Working Notes*, 950–965.

Chen, Ziyi, Ren Yang, Sunyang Fu, Nansu Zong, Hongfang Liu & Ming Huang. 2023. Detecting Reddit users with depression using a hybrid neural network SBERT-CNN. Em *IEEE 11th International Conference on Healthcare Informatics (ICHI)*, 193–199. 10.1109/ICHI57859.2023.00035

Chiong, Raymond, Gregorius Satia Budhi, Sandeep Dhakal & Fabian Chiong. 2021. A textual-based featuring approach for depression detection using machine learning classifiers and social media texts. *Computers in Biology and Medicine* 135. 104499. 10.1016/j.compbiomed.2021.104499

Cohan, Arman, Bart Desmet, Andrew Yates, Luca Soldaini, Sean MacAvaney & v Goharian. 2018. SMHD: a large-scale resource for exploring online language usage for multiple mental health conditions. Em *27th International Conference on Computational Linguistics (COLING)*, 1485–1497

Coppersmith, Glen, Mark Dredze, Craig Harman, Hollingshead Kristy & Margaret Mitchell. 2015. CLPsych 2015 shared task: Depression and PTSD on Twitter. Em *2nd Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 31–39. 10.3115/v1/W15-1204

da Costa, Pablo Botton, Matheus Camasmie Pavan, Wesley Ramos dos Santos, Samuel Caetano da Silva & Ivandré Paraboni. 2023. BERTabaporu: assessing a genre-specific language model for Portuguese NLP. Em *Recents Advances in Natural Language Processing (RANLP)*, 217–223. 10.26615/978-954-452-092-2_024

de Cássia Alves, Vanessa, Franco Eusébio Garcia, Conrado Saud, Augusto Mendes, Helena Medeiros Caseli, Vivian Genaro Motti, Luciano de Oliveira Neris, Tais Bleicher & Vânia P. Almeida Neris. 2024. College students-in-the-loop for their mental health: a case of ai and humans working together to support well-being. *Interaction Design & Architecture(s)* 59. 79–94. 10.55612/s-5002-059-003

Dattani, Saloni, Hannah Ritchie & Max Roser. 2021. Mental health: Our world in data. [Online].

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 4171–4186. 10.18653/v1/N19-1423

Eichstaedt, Johannes C., Robert J. Smith, Raina M. Merchant, Lyle H. Ungar, Patrick Crutchley, Daniel Preoţiuc-Pietro, David A. Asch & H. Andrew Schwartz. 2018. Facebook language predicts depression in medical records. *Proceedings of the National Academy of Sciences* 115(44). 11203–11208. 10.1073/pnas.1802331115

Fabregat, Hermenegildo, Andres Duque, Lourdes Araujo & Juan Martinez-Romo. 2022. UNED-NLP at eRisk 2022: Analyzing gambling disorders in social media using approximate nearest neighbors. Em *CLEF 2022 Working Notes*, 894–904.

Fabregat, Hermenegildo, Andres Duque, Lourdes Araujo & Juan Martinez-Romo. 2023. NLP-UNED-2 at eRisk 2023: detecting pathological gambling in social media through dataset relabeling and neural networks. Em *CLEF 2023 Working Notes*, 672–683.

Ferreira, Rodrigo, Alina Trifan & José Luís Oliveira. 2022. Early risk detection of mental illnesses using various types of textual features. Em *CLEF 2022 Working Notes*, 905–920.

Flores, Arthur Marçal, Matheus Camasmie Pavan & Ivandré Paraboni. 2022. User profiling and satisfaction inference in public information access services. *Journal of Intelligent Information Systems* 58(1). 67–89. 10.1007/s10844-021-00661-w

Gamoran, Avi, Yonatan Kaplan, Almog Simchon & Michael Gilead. 2021. Using psychologically-informed priors for suicide prediction in the CLPsych 2021 shared task. Em *7th Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 103–109. 10.18653/v1/2021.clpsych-1.12

Gollapalli, Sujatha Das, Guilherme Augusto Zagatti & See-Kiong Ng. 2021. Suicide risk prediction by tracking self-harm aspects in Tweets: NUS-IDS at the CLPsych 2021 shared task. Em *7th Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 93–98. 10.18653/v1/2021.clpsych-1.10

Larrayoz, Xabier, Nuria Lebeña, Arantza Casillas & Alicia Pérez. 2023. Representation exploration and deep learning applied to the early detection of pathological gambling risks. Em *CLEF 2023 Working Notes*, 693–705.

Lestandy, Merinda, Amrul Faruq, Adhi Nugraha et al. 2024. Analyzing reddit data: Hybrid model for depression sentiment using FastText embedding. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)* 8(2). 288–297. 10.29207/resti.v8i2.5641

Lopes, Rui Pedro. 2021. CeDRI at eRisk 2021: A naive approach to early detection of psychological disorders in social media. Em *CLEF 2021 Working Notes*, 981–991.

Losada, David E., Fabio Crestani & Javier Parapar. 2017. eRISK 2017: CLEF lab on early risk prediction on the internet: experimental foundations. Em *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 346–360. 10.1007/978-3-319-65813-1_30

Losada, David E., Fabio Crestani & Javier Parapar. 2018. Overview of eRisk: Early risk prediction on the internet. Em *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 343–361. 10.1007/978-3-319-98932-7_30

Losada, David E., Fabio Crestani & Javier Parapar. 2019. Overview of eRisk 2019 early risk prediction on the internet. Em *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 340–357. 10.1007/978-3-030-28577-7_27

Losada, David E, Fabio Crestani & Javier Parapar. 2020. Overview of eRisk at CLEF 2020: Early risk prediction on the internet (extended overview). Em *CLEF 2020 Working Notes*,

Loyola, Juan Martín, Sergio Burdisso, Horacio Thompson, Leticia Cagnina & Marcelo Errecalde. 2021. UNSL at eRisk 2021: A comparison of three early alert policies for early risk detection. Em *CLEF 2021 Working Notes*, 992–1021.

Loyola, Juan Martín, Horacio Thompson, Sergio Burdisso & Marcelo Errecalde. 2022. UNSL at eRisk 2022: Decision policies with history for early classification. Em *CLEF 2022 Working Notes*, 947–960.

Lynn, Veronica, Alissa Goodman, Kate Niederhoffer, Kate Loveys, Philip Resnik & H. Andrew Schwartz. 2018. CLPsych 2018 Shared Task: Predicting current and future psychological health from childhood essays. Em *5th Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 37–46. 10.18653/v1/W18-0604

Mann, Paulo, Aline Paes & Elton H. Matsushima. 2020. See and read: Detecting depression symptoms in higher education students using multimodal social media data. Em *40th International AAAI Conference on Web and Social Media*, vol. 1 14, 440–451. 10.1609/icwsm.v14i1.7313

Mármol-Romero, Alba María, Salud María Jiménez-Zafra, Flor Miriam Plaza-del Arco, M Dolores Molina-González, María-Teresa Martín-Valdivia & Arturo Montejo-Ráez. 2022. SINAI at eRisk@ CLEF 2022: Approaching early detection of gambling and eating disorders with natural language processing. Em *CLEF 2022 Working Notes*,

Mármol-Romero, Alba María, Adrián Moreno-Muñoz, Flor Miriam Plaza-del Arco, María Dolores Molina-González, Maria Teresa Martín-Valdivia, Luis Alfonso Ureña-López & Arturo Montejo-Raéz. 2023. Overview of MentalriskES at IberLEF 2023: Early detection of mental disorders risk in Spanish. *Procesamiento del Lenguaje Natural* 71. 329–350. 10.26342/2023-71-26

Martínez-Castano, Rodrigo, Amal Htait, Leif Azzopardi & Yashar Moshfeghi. 2020. Early risk detection of self-harm and depression severity using BERT-based transformers. Em *CLEF 2020 Working Notes*,

Masood, Razan, Faneva Ramiandrisoa & Ahmet Aker. 2019. UDE at Erisk 2019: early risk prediction on the internet. Em *CLEF 2019 Working Notes*,

Maupomé, Diego, Maxime D. Armstrong, Fanny Rancourt, Thomas Soulas & Marie-Jean Meurs. 2021. Early detection of signs of pathological gambling, self-harm and depression through topic extraction and neural networks. Em *CLEF 2021 Working Notes*, 1031–1045.

Mendes, Augusto R. & Helena M. Caseli. 2024. Identifying fine-grained depression signs in social media posts. Em *Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*, 8594–8604.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado & Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. Em *Advances in Neural Information Processing Systems*, vol. 26,

Mohammadi, Elham, Hessam Amini & Leila Kosseim. 2019. Quick and (maybe not so) easy detection of anorexia in social media posts. Em *CLEF 2019 Working Notes*,

Molina, Antonio, Xinhui Huang, Lluís-F Hurtado & Ferran Pla. 2023. ELiRF-UPV at eRisk 2023: Early detection of pathological gambling using SVM. Em *CLEF 2023 Working Notes*, 736–742.

Morales, Michelle, Prajjalita Dey & Kriti Kohli. 2021. Team 9: A comparison of simple vs. complex models for suicide risk assessment. Em *7th Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 99–102. 10.18653/v1/2021.clpsych-1.11

Naderi, Nona, Julien Gobeill, Douglas Teodoro, Emilie Pasche & Patrick Ruch. 2019. A baseline approach for early detection of signs of anorexia and self-harm in Reddit posts. Em *CLEF 2019 Working Notes*,

Parapar, Javier, Patricia Martín-Rodilla, David E. Losada & Fabio Crestani. 2021. Overview of eRisk 2021: Early risk prediction on the internet. Em *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 324–344. 10.1007/978-3-030-85251-1_22

Parapar, Javier, Patricia Martin-Rodilla, David E. Losada & Fabio Crestani. 2022. Overview of eRisk 2022: Early risk prediction on the internet. Em *CLEF 2022 Working Notes*, 821–850.

Parapar, Javier, Patricia Martín-Rodilla, David E Losada & Fabio Crestani. 2023. Overview of eRisk 2023: Early risk prediction on the internet. Em *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, 294–315. 10.1007/978-3-031-42448-9_22

Pavan, Matheus Camasmie, Vitor Garcia dos Santos, Alex Gwo Jen Lan, Jo ao Trevisan Martins, Wesley Ramos dos Santos, Caio Deutsch, Pablo Botton da Costa, Fernando Chiu Hsieh & Ivandré Paraboni. 2023. Morality classification in natural language text. *IEEE transactions on Affective Computing* 14(1). 857–863. 10.1109/TAFFC.2020.3034050

Pennebaker, James W., Martha E. Francis & Roger J. Booth. 2001. *Linguistic inquiry and word count: LIWC*. Lawrence Erlbaum

Ragheb, Waleed, Jérôme Azé, Sandra Bringay & Maximilien Servajean. 2019. Attentive multi-stage learning for early risk detection of signs of anorexia and self-harm on social media. Em *CLEF 2019 Working Notes*,

Sadeque, Farig, Dongfang Xu & Steven Bethard. 2018. Measuring the latency of depression detection in social media. Em *11th ACM International Conference on Web Search and Data Mining*, 495–503. 10.1145/3159652.3159725

dos Santos, Wesley Ramos, Amanda Maria Martins Funabashi & Ivandré Paraboni. 2020. Searching Brazilian Twitter for signs of mental health issues. Em *12th International Conference on Language Resources and Evaluation (LREC)*, 6113–6119.

dos Santos, Wesley Ramos, Rafael Lage de Oliveira & Ivandré Paraboni. 2023. SetembroBR: a social media corpus for depression and anxiety disorder prediction. *Language Resources and Evaluation* 273–300. 10.1007/s10579-022-09633-0

dos Santos, Wesley Ramos & Ivandré Paraboni. 2023. Predição de transtorno depressivo em redes sociais: BERT supervisionado ou ChatGPT zero-shot? Em *14th Simposio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, 11–21. 10.5753/stil.2023.233275

dos Santos, Wesley Ramos & Ivandré Paraboni. 2024. Prompt-based mental health screening from social media text. Em *13th Brazilian Workshop on Social Network Analysis and Mining*, 186–192. 10.5753/brasnam.2024.1879

Sauberli, Andreas, Sooyeon Cho & Laura Stahlhut. 2022. LauSAn at eRisk 2022: Simply and effectively optimizing text classification for early detection. Em *CLEF 2022 Working Notes*, 995–1004.

da Silva, Samuel Caetano, Thiago Castro Ferreira, Ricelli Moreira Silva Ramos & Ivandré Paraboni. 2020. Data driven and psycholinguistics motivated approaches to hate speech detection. *Computación y Systemas* 24(3). 1179–1188. 10.13053/CyS-24-3-3478

de Souza, Vanessa Borba, Jeferson Campos Nobre & Karin Becker. 2022. DAC stacking: A deep learning ensemble to classify anxiety, depression, and their comorbidity from reddit texts. *IEEE Journal of Biomedical and Health Informatics* 26(7). 3303–3311. 10.1109/jbhi.2022.3151589

Srivastava, Harshvardhan, Lijin N. S, Sruthi S & Tanmay Basu. 2022. NLP-IISERB@eRisk2022: Exploring the potential of bag of words, document embeddings and transformer based framework for early prediction of eating disorder, depression and pathological gambling over social media. Em *CLEF 2022 Working Notes*, 972–986.

Tao, Yongfeng, Minqiang Yang, Hao Shen, Zhichao Yang, Ziru Weng & Bin Hu. 2023. Classifying anxiety and depression through LLMs virtual interactions: A case study with ChatGPT. Em *IEEE International Conference on Bioinformatics and Biomedicine*, 2259–2264. 10.1109/BIBM58861.2023.10385305

Thompson, Horacio, Leticia Cagnina & Marcelo Errecalde. 2023. Strategies to harness the transformers' potential: UNSL at eRisk 2023. Em *CLEF 2023 Working Notes*, 791–804.

Trifan, Alina & José Luís Oliveira. 2019. BioInfo@UAVR at eRisk 2019: delving into social media texts for the early detection of mental and food disorders. Em *CLEF 2019 Working Notes*,

Wang, Ning, Fan Luo, Yuvraj Shivtare, Varsha D Badal, KP Subbalakshmi, Rajarathnam Chandramouli & Ellen Lee. 2021. Learning models for suicide prediction from social media posts. ArXiv [cs.CL]. 10.48550/arXiv.2105.03315

Zafar, Aqsa & Sanjay Chitnis. 2020. Survey of depression detection using social networking sites via data mining. Em *10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 88–93. 10.1109/Confluence47617.2020.9058189

Artigos Técnicos

Proposta recebida em Julho 2024 e aceite para publicação em Outubro 2024.

Aperfeiçoando a Hifenização Automática em Português no TEX

Enhancing Automatic Hyphenation in Portuguese for TEX

Leonardo Araujo
Universidade Federal de São João del-Rei

Aline Benevides
Faculdade de Tecnologia do Estado de São Paulo

Resumo

As regras de hifenização do português para o TEX têm sido utilizadas há mais de três décadas, apresentando um bom desempenho geral. Entretanto, ainda há hifenizações incorretas e pontos de hifenização não identificados. Esses pontos, embora em sua maioria ocorram perto das bordas das palavras e sejam irrelevantes para fins tipográficos no TEX, podem ser relevantes em contextos específicos, como ao lidar com palavras fora do léxico padrão ou com aplicações que fazem o uso da segmentação silábica/tipográfica. A partir de uma análise de 49.528 palavras hifenizadas, obtidas de dicionários online, propusemos 120 novas regras para serem incorporadas às regras existentes de hifenização do português. Além disso, utilizamos o *patgen* para criar novas regras ou melhorar as já existentes. No entanto, as regras geradas pelo *patgen* não demonstraram boa capacidade de generalização. Em última análise, as regras manuais ajustadas apresentaram o melhor desempenho, resultando em um aumento de 2.1% na taxa de acertos. O número de pontos de hifenização corretos aumentou de 38.519 para 39.808, enquanto os pontos de hifenização incorretos diminuíram drasticamente de 2.059 para 33. Importante ressaltar também que as regras elaboradas manualmente demonstraram uma melhor capacidade de generalização do que as regras geradas automaticamente pelo *patgen*.

Palavras chave

hifenização; padrões de hifenização; hifenização automática em Português

Abstract

Portuguese hyphenation rules for TEX have been in use for over three decades, showing good overall performance. However, there are still incorrect hyphenations and undetected hyphenation points. These points, although mostly occurring near word boundaries and being irrelevant for typographic purposes in TEX, can be relevant in specific contexts, such as when dealing with words outside the standard lexicon or in applications that utilize syllabic/typographic segmentation. Based on an analysis of 49,528 hyphenated words obtained from online dictionaries, we proposed 120 new rules to be incorporated into the existing Portuguese hyphenation rules. Additionally, we used *patgen* to create new rules or improve existing ones. However, the rules generated by *patgen* did not demonstrate good generalization capability. Ultimately, the manually adjusted rules showed the best performance, resulting in a 2.1% increase in the success rate. The number of correct hyphenation points increased from 38,519 to 39,808, while the incorrect hyphenation points drastically decreased from 2,059 to 33. It is also important to note that the manually crafted rules demonstrated better generalization capability than the automatically generated rules by *patgen*.

Keywords

hyphenation; hyphenation patterns; automatic hyphenation in Portuguese

1. Introdução

Durante muito tempo, o hífen não era utilizado na quebra de linha, de forma que as palavras precisavam caber inteiramente em uma linha ou eram cortadas em locais arbitrários. Inicialmente, nenhuma marcação era utilizada para indicar a quebra de linha, o que poderia causar confusão ou alguma interpretação indesejada. Como resultado, os ortógrafos advogaram pela introdução de um sinal para indicar essas quebras. O português enfrentou a mesma introdução gradual de um sinal de hifenização para marcar palavras que se estendiam por linhas. Embora o uso de um sinal de hifenização (=) tenha sido defendido pelos ortógrafos (Gândavo, 1981), poucos documentos utilizavam esse sinal até o final do século XVIII (de Araújo & Maruyama, 2015).

A hifenização pode dificultar a leitura fluida e, por isso, em alguns casos deve ser evitada. Na literatura infantil, por exemplo, pode interromper o fluxo e a compreensão da leitura. Da mesma forma, em fontes grandes, títulos ou manchetes, a hifenização parece visualmente desagradável. Em documentos técnicos, pode causar confusão quando aplicada a jargões. Por outro lado, espaços grandes ou pequenos entre pala-

DOI: 10.21814/lm.16.2.435
This work is Licensed under a Creative Commons Attribution 4.0 License

vras também podem dificultar o processo de leitura, tornando a hifenização fundamental quando os textos usam comprimentos de linha curtos. À medida que uma linha fica mais curta, o número de candidatos à quebra entre as palavras diminui, levando a espaços irregulares entre as palavras e os caracteres. Em textos alinhados à esquerda, o espaçamento consistente pode eliminar o risco de formação de rios (espaços brancos irregulares), garantindo uma aparência consistente e esteticamente agradável do texto. Da mesma forma, em texto justificado, a hifenização pode criar espaçamento irregular, reduzindo a legibilidade. Portanto, a hifenização automática desempenha um papel importante na tipografia.

O TEX, como sistema de composição tipográfica, aborda meticulosamente essas questões. Através da hifenização automática e do ajuste fino do espaçamento entre pares específicos de letras (chamado de *kerning*), ele organiza o texto na página para proporcionar uma leitura fluida e visualmente agradável. Esse processo considera diversos fatores que influenciam a legibilidade, como altura e comprimento das linhas, tamanho dos parágrafos, tipo e tamanho da fonte, e espaçamento entre letras e palavras.

O espaçamento ideal entre palavras evita a formação de 'lagos' (espaços excessivos) e 'rios' (colunas irregulares de espaços brancos), garantindo clareza e conforto visual. Além disso, o TEX previne 'órfãos' (última linha de um parágrafo com apenas uma ou poucas palavras) e 'viúvas' (primeira linha de um parágrafo sozinha na página anterior), melhorando o fluxo do texto. O controle automático dos espaços em branco entre caracteres, palavras, linhas, parágrafos, seções e recuos proporciona um espaçamento ideal entre os elementos do texto, garantindo uma distribuição uniforme na página, evitando problemas de legibilidade e estética.

O presente trabalho se propõe a analisar as regras de hifenização automática do TEX, revisitar a utilização do programa *patgen* para criar novas regras, e analisar as regras de hifenização do português, comparando aquelas propostas por de Rezende (1987); de Rezende & Almeida (2015) e as atualizações propostas por Araujo & Benevides (2024). Para realizar tal comparação, será utilizado como referência um dicionário de hifenização do português criado a partir de seis recursos lexicais online Michaelis (sd.), Priberam (sd.), Wikcionário (sd.), Aulete (sd.), Portal da Língua Portuguesa (sd.) e Dicio (sd.).

Araujo & Benevides (2024) propõem a atualização das regras de hifenização para o TEX, utilizando regras elaboradas manualmente e comparando-as com aquelas geradas pelo patgen, destacando a limitada capacidade de generalização deste último. O presente artigo aprofunda e amplia o escopo da análise anterior, abordando o tema sob uma perspectiva mais abrangente da língua portuguesa, com foco nos aspectos fonológicos, gramaticais e estilísticos. Além de esclarecer as escolhas adotadas, o estudo aponta direções para futuras melhorias e discute as limitações inerentes ao processo de hifenização automática.

2. Hifenização

Os idiomas, em geral, podem ter suas hifenizações guiadas de quatro critérios distintos ou pela relação entre elas: (1) fonológico, baseada na divisão silábica da fala; (2) morfológico (ou etimológico), focada nas partes que proveem significado às palavras (prefixos, radicais e sufixos); (3) ortográfico, seguindo convenções de escrita padrão; e (4) semântico, que considera o contexto para evitar cortes ambíguos ou inadequados.

A regra geral de hifenização no português é dividir nas fronteiras entre sílabas. Uma sílaba é formada por um núcleo obrigatório, que, no português, pode ser preenchido apenas por vogal, e consoantes periféricas opcionais (antes ou depois do núcleo). Em algumas situações, a divisão silábica não respeita os constituintes morfológicos, criando uma relação conflitante com o padrão morfológico, que também desempenha papel fundamental na hifenização no português. Prefixos como *bis-*, *des-* e *in-* são exemplos: de um lado, temos hifenizações, como *bis-ne-to*, *des-co-brir* e *in-va-ri-á-vel*, em que as fronteiras morfológicas são respeitas; de outro lado, temos *bi-sa-vô*, *de-sem-bar-que*, *i-na-ti-vo* e *i-nobs-tan-te*[1], em que os prefixos são divididos em duas sílabas—neste caso, a hierarquia de sonoridade dita a preferência por preencher o ataque sílaba, e não a coda.

Em alguns casos, a hifenização também é uma questão de estilo, tendo em vista que algumas divisões soam melhores que outras. Essas alternativas conflitantes geralmente surgem quando uma palavra possui muitos pontos de hifenização possíveis. Considere *ar-que-ó-lo-go*, que é preferencialmente particionado como *arque-ólogo* em oposição a *arqueó-logo*. É preferível manter morfemas inteiros juntos: *arque* (que significa an-

[1] A regra de divisão silábica pode levar a duas partições possíveis, baseadas em forças fonológicas ou morfológicas: *i-nobs-tan-te* e *in-obs-tan-te*, mas a primeira é preferível.

tigo, primitivo) e *ólogo* (aquele que estuda o assunto). Em português, a palavra *en-tres-sa-fra* é preferencialmente separada como *entres-safra* em oposição a *en-tressafra* ou *entressa-fra*, e a palavra *fe-liz-men-te* é preferencialmente separada como *feliz-mente* em oposição a *fe-lizmente* ou *felizmen-te*. Em ambos os casos, manter os dois morfemas juntos foi a principal motivação. Também é mais elegante evitar divisões entre consoantes ou vogais duplas, mesmo que exista um ponto de hifenização entre essas letras. Por exemplo, *pressu-rizar* é preferível a *pres-surizar* e *empreen-dedor* é preferível a *empre-endedor*. Ainda assim, existem exceções, sendo preferível particionar *micro-organismo* em vez de *microorganismo* (manter morfemas juntos é preferível a dividir uma vogal dupla).

A hifenização ortográfica considera como as palavras são escritas e segue as convenções de um determinado idioma. Em português, a hifenização evita separar dígrafos quando formados por consoantes distintas, como *lh* em *mi-lho* e *nh* em *ba-nho*; mas as separam quando constituídas por consoantes iguais, como *ss* em *as-sim* e *rr* em *ter-ra*. Também tende a manter certos agrupamentos consonantais juntos, como *br* em *cobra* (*co-bra*) e *tr* em *atração* (*a-tra-ção*). Há, ainda, casos como *bíceps* (*bí-ceps*) e *advogado* (*ad-vo-ga-do*), com consoantes mudas na grafia, mas com epêntese na fala. As sequências consonantais constituiriam sílabas próprias, entretanto a ortografia impede que sejam separadas deixando uma sequência não permitida pela fonotática da língua (*bí-ce-ps* e *a-d-vo-ga-do*, respectivamente).

Uma palavra com diferentes significados pode ser hifenizada de formas distintas de acordo com o sentido. Por exemplo, a palavra sueca *glassko* possui três significados e pode ser hifenizada como *glas-sko* (sapato de vidro), *glass-ko* (vaca de sorvete) e, de forma não padrão, *glass-sko* (sapato de sorvete) (Németh, 2006). Em português, a palavra *sublinha* pode ser hifenizada de duas maneiras: *su-bli-nha* (verbo) quando representa a forma flexionada do verbo *sublinhar* ou *sub-linha* (substantivo) quando se refere ao traço sublinhado.

Outro ponto importante a considerar são as ambiguidades que podem surgir quando uma palavra é particionada durante o processo de hifenização. No âmbito da composição tipográfica, em português, devemos evitar hifenizações como *de-putada*, *fede-ração*, *acu-mula*, *após-tolo* e *cú-bico*. Hifenizações que possam levar o leitor a pronunciar incorretamente uma palavra também devem ser evitadas, como *pe-rigo*.

Diante disso, a automatização desse processo pode se dar de duas maneiras: i. a partir de uma abordagem baseada em dicionário, que restringe as possibilidades de hifenização às entradas presentes em um dicionário; ou ii. a partir de uma abordagem baseada em algoritmo, que pode ser aplicada a qualquer sequência encontrada no texto. Um modelo baseado em regras pode incluir o reconhecimento de prefixos, sufixos, morfemas ou sequências adequadas para a inserção de um hífen. A abordagem baseada em algoritmo pode utilizar um sistema lógico para analisar palavras e aplicar as regras de hifenização de um determinado idioma. Como as regras de hifenização variam significativamente entre línguas, é necessário desenvolver um algoritmo para cada uma. Embora um sistema baseado em lógica possa ser eficiente e compacto, ele ainda precisa lidar com exceções por meio de regras pré-programadas. Outra abordagem é o uso de padrões de correspondência. Ao utilizar um corpus com exemplos de hifenização em um idioma, essa abordagem identifica sequências de letras que determinam pontos de hifenização adequados. Os padrões podem englobar prefixos, sufixos, exceções e regras especiais de hifenização do idioma. Por fim, um modelo híbrido também pode ser utilizado. Ele combina dois ou mais dos métodos descritos anteriormente para aprimorar a precisão e flexibilidade da hifenização.

3. Hifenização automática no TEX

A hifenização automática é um componente crucial dos sistemas de preparação de documentos, especialmente na composição tipográfica e na editoração eletrônica. Ela contribui para a legibilidade e estética do texto. Ao automatizar a inserção de hifens em locais adequados, dispensa-se a intervenção de editores ou tipógrafos e, ainda, pode ser facilmente adaptada para diferentes idiomas.

Apesar do fato de o algoritmo e as regras de hifenização do TEX serem antigos, eles são, até hoje, a abordagem mais usada, mesmo fora do mundo do TEX. A base para isso é o Hunspell, um verificador ortográfico e analisador morfológico adotado por diversos softwares (por exemplo, LibreOffice, Mozilla Firefox, Mozilla Thunderbird, Google Chrome, macOS, InDesign, memoQ, Opera, Affinity Publisher, entre outros (Hunspell's Team, 2023)). O Hunspell usa as regras de hifenização do TEX (Hunspell's Team, sd.; Levien, 1998), tornando a hifenização do TEX amplamente difundida no mundo da computação. Isso é resultado da simplicidade e versatilidade da abordagem do TEX. O algoritmo funciona de maneira eficaz, pois já suporta re-

gras para 66 idiomas (TEX pattern authors, sd.) e oferece flexibilidade para criar regras para qualquer idioma atualmente não suportado.

A hifenização automática no TEX foi introduzida por Knuth (1977). O seu desenvolvimento focava nas regras para o inglês, as quais compreendiam: (1) cada parte deve conter uma vogal, exceto *e* final; (2) remoção de sufixo; (3) remoção de prefixo; e (4) quebra vogal-consoante-consoante-vogal (VCCV) (mas combinando *h* com a letra anterior se for consoante). Testes mostraram que este primeiro algoritmo poderia encontrar 40% dos pontos de hifenização permitidos (Liang, 1983). O algoritmo de hifenização proposto por Liang (1983) e adotado no TEX usa a noção de padrões concorrentes (Liang, 1983). Um banco de dados de palavras hifenizadas é varrido em busca de padrões de hifenização e inibição. O algoritmo introduzido no TEX82 utiliza cinco níveis alternados de padrões de hifenização e inibição. O *patgen*, programa para geração de padrões baseado em um corpus, foi criado por Liang (1983) (Liang & Breitenlohner, 1991) e usado para criar padrões de hifenização para vários idiomas (Sojka, 1995a,b; Sojka et al., 2005; Sojka & Antoš, 2003; Scannell, 2003).

A hifenização efetiva de palavras pelo TEX depende dos seguintes fatores: (1) idioma do documento, que determina qual conjunto de padrões aplicar; (2) caracteres usados, pois alguns podem bloquear a hifenização em suas extremidades; (3) o valor das variáveis internas `\lefthyphenmin` e `\righthyphenmin`, que definem o comprimento mínimo da sequência de caracteres nas bordas esquerda e direita antes que qualquer hifenização seja permitida (Sojka, 2002).

Para simplificar, se considerarmos apenas o alfabeto latino, sem caracteres diacríticos, os padrões usados na hifenização do TEX têm a forma: `^\.?[0-9]?([a-z]+[0-9]?)+\.?\$`, sendo esta uma expressão regular.Um exemplo desse padrão é `4z1z2` (parte das regras de hifenização do inglês), que é composto por uma sequência de letras e números. Em geral, usamos caracteres/símbolos do alfabeto da língua, junto com números indo-arábicos, para expressar facilitação ou inibição de hifenização. Números ímpares indicam um bom ponto de hifenização, enquanto números pares indicam um local inadequado para quebra. O exemplo dado indica que a sequência tem um bom ponto de quebra entre o primeiro e o segundo *z* e que a hifenização deve ser inibida antes do primeiro *z* e após o segundo *z*. Por exemplo, a hifenização das palavras *piz-za*, *fiz-zle* e *mezzanine* segue essa regra, em que o hífen é colocado entre os dois *z* e nenhum hífen antes nem depois dos *z*. Os padrões também podem usar o ponto (.) para indicar limites de palavras. O padrão `.sh2` se aplica ao início das palavras, o que implica que o *s* e o *h* devem ficar juntos no início de uma palavra e a hifenização também deve ser inibida após o *h*. Por exemplo, este padrão é utilizado em *Sher-lock*.

As regras de hifenização são organizadas em níveis, de 1 a 9, onde números ímpares representam níveis de hifenização e números pares representam níveis de inibição. Cada nível funciona como um nível de exceção para o seu predecessor. Por exemplo, a regra `sh1er` indica um bom ponto de hifenização entre o *h* e o *e* na sequência *sher*. Uma regra em um nível superior, como `.sh2`, implica uma exceção à regra do nível inferior. Quando vemos *sher* no início de uma palavra, a regra `.sh2` se aplica e a hifenização proposta pela regra do nível inferior `sh1er` deve ser impedida. Esse é o caso da hifenização da palavra *Sher-lock*. O exemplo completo é fornecido na Listagem 1[2], na qual é possível verificar todas as regras pertinentes do inglês atuando na hifenização de *Sher-lock*.

```
.  s  h  e  r  l  o  c  k  .
   0  0  2  |  |  |  |  |  |  .sh2
   0  2  0  |  |  |  |  |  |  s2h
   0  0  1  0  0  |  |  |  |  sh1er
   |  |  |  0  1  0  |  |  |  r1l
   |  |  |  0  3  0  4  |  |  r3lo4
   |  |  |  |  |  |  0  0  1  ck1
max: 0  2  2  0  3  0  4  0  1
final: s  h  e  r  -  l  o  c  k  -
```

Listagem 1: Exemplo das regras aplicadas à hifenização da palavra *Sherlock*.

Algumas regras de hifenização não podem ser implementadas usando o algoritmo de hifenização do TEX, tendo em vista que, no alemão, por exemplo, a hifenização pode levar à alteração ou à inserção de letras. Além disso, palavras compostas não possuem hifens, resultando em sequências longas de letras sem separação visível e até mesmo repetições da mesma letra, como visto em *Wasserrinne* e *Schifffahrt*. A reforma ortográfica do alemão também introduziu algumas mudanças, tornando necessária a criação de um conjunto diferente de regras de hifenização. Por exemplo, a palavra *Schiffahrt* deve ser hifenizada como *Schiff-fahrt*, preservando os *f*s de cada palavra que constituem esse composto. A hifenização deve inserir um *f* que não faz parte da forma escrita, o que não era um problema para a antiga forma escrita da palavra: *Schifffahrt*. Além disso, as antigas regras de hife-

[2]Este exemplo foi criado utilizando-se o algoritmo de hifenização do TEX portado para a linguagem Go. Seu código está disponível em https://github.com/speedata/hyphenation.

nização da gramática alemã estabeleciam a hifenização *Bäk-ker* para a palavra *Bäcker*, *Zuk-ker* para a palavra *Zucker* e *pak-ken* para a palavra *packen*. Atualmente, essas palavras são hifenizadas como *Bä-cker*, *Zu-cker* e *pa-cken*, respectivamente. Algumas palavras também mudaram seu ponto de hifenização após a reforma ortográfica: por exemplo, *Fen-ster* tornou-se *Fens-ter* e *meistens* tornou-se *meis-tens*. Alguns problemas na hifenização de palavras compostas no TEX são discutidos em Sojka (1995b).

Em síntese, um padrão consiste em uma sequência de caracteres (do alfabeto da linguagem) podendo conter um número intercalado, expressando o nível de hifenização/inibição. O marcador de limite de palavra, isto é, o ponto final, pode ser usado ocasionalmente nas extremidades do padrão. Quando não há número entre os caracteres em um padrão, assume-se o valor zero, o que significa *indefinido* e nenhum ponto de hifenização será sugerido nessa localização.

3.1. Patgen

Patgen utiliza uma lista de palavras hifenizadas para extrair padrões e, a partir deles, definir regras em vários níveis e com diferentes comprimentos. Ele começa com padrões curtos e aumenta incrementalmente seu comprimento até atingir o comprimento máximo permitido pelo usuário. O objetivo é manter os padrões o mais concisos possível, para obter uma melhor generalização. À medida que avança e padrões mais longos são incorporados, o *patgen* estabelece exceções. Em certos casos, pode ser necessária a análise de padrões longos, pois alguns pontos de hifenização podem depender de caracteres distantes do ponto de quebra.[3]

O *Patgen* trabalha com índices de *glifo*[4] em vez de códigos de caracteres. Cada glifo é representado por um único byte. Isso resulta em 256 índices, onde 13 deles são reservados para os dígitos 0-9 e os caracteres '.', '-' e '*'. Os 243 restantes são usados para representar símbolos de uma determinada língua. Os dígitos 0-9 são reservados para expressar níveis de regras de hifenização, e os caracteres '.', '-' e '*' são reservados para expressar um ponto de hifenização incorreto, um ponto de hifenização ausente e um ponto de hifenização correto, respectivamente. Para executar o *patgen*, é necessário um arquivo de tradução. Este arquivo define os valores de certos parâmetros específicos do idioma (na primeira linha) e enumera as várias formas como os símbolos do idioma podem aparecer (todas as linhas subsequentes). Na primeira linha, as posições 1 e 2 são usadas para definir o valor de `lefthyphenmin`, e as posições 3 e 4 são usadas para definir o valor de `righthyphenmin`. Esses valores determinam o comprimento mínimo de uma sequência de caracteres que pode ser gerada por um procedimento de hifenização. Para definir um valor de um único dígito, deixe a primeira posição em branco, ou seja, coloque um espaço nas posições 1 e 3 para `lefthyphenmin` e `righthyphenmin`, respectivamente. As posições 5, 6 e 7 são usadas para definir valores alternativos para os caracteres especiais '.', '-' e '*'. Um esquemático da primeira linha do arquivo de tradução é apresento na Figura 1.

As linhas a seguir usam um delimitador para demarcar cada letra do alfabeto da linguagem desejada, incluindo suas representações alternativas. A primeira posição da linha define o delimitador, e cada símbolo da linguagem pode ocupar tantas posições quanto necessárias, desde que o valor reservado para o delimitador não seja usado na definição do símbolo. As linhas de definição terminam quando o delimitador aparece duas vezes consecutivas. Considere o seguinte exemplo para definir a letra *e*, fazendo-se a suposição de equivalência entre as várias formas em que tal letra pode aparecer (minúsculas, maiúsculas, com ou sem diacríticos):

```
XeXEXéXêXÉXÊX\'{e}X\^{e}X\'{E}X\^{E}XX.
```

Este exemplo está ilustrado na Figura 2 para uma melhor compreensão. Adotamos `X` como delimitador. Observe que usamos a entrada direta (usando UTF-8 ou outra codificação que suporte o caractere *e* acentuado) e também a contraparte composicional usando a sequência de controle TEX apropriada que instrui o TEX a colocar diacrítico no caractere. Nesta definição, assumimos que as várias formas nas quais podemos encontrar o caractere *e* serão equivalentes para fins de hifenização (padrão de correspondência). Vale ressaltar que não é este o caso do português, o exemplo tem caráter meramente ilustrativo. O acento, muitas vezes representado graficamente pela presença de diacrítico, possui papel preponderante na determinação de sílabas e, consequentemente, na hifenização no português. Como outro exemplo, veja a linha que define o caractere π (Haralambous, 2021): `#p#P#\varpi ##.`

[3] Alguns exemplos, em inglês, de dependência de hifenização em caracteres distantes do ponto de quebra são: *dem-o-crat* e *de-moc-ra-cy*; *as-pi-rin* e *aspir-ing*; *de-monstra-tive* e *dem-on-stra-tion*.

[4] O termo *glifo* é usado comumente em linguística, tipografia e computação gráfica para se referir a uma representação gráfica específica de um caractere ou símbolo, que pode ser o símbolo inteiro ou um elemento visual distinto dentro dele.

1	2	3	4	5	6	7
lefthyphenmin		righthyphenmin		·	-	*

Figura 1: Primeira linha do arquivo de tradução utilizado pelo *patgen*.

	delimitador	letra e	delimitador	letra E	delimitador	letra é	delimitador	letra ê	delimitador	letra É	delimitador	letra Ê	delimitador	sequência TEX para é	delimitador	sequência TEX para ê	delimitador	sequência TEX para É	delimitador	sequência TEX para Ê
1	2		3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
X	e		X	E	X	é	X	ê	X	É	X	Ê	X	\'{e}	X	\^{e}	X	\'{E}	X	\^{E}

Figura 2: Exemplo de uma linha do arquivo de tradução onde são definidas as várias formas equivalentes para a letra *e*.

O *Patgen* também necessita de um arquivo de dicionário, que é uma lista de palavras pré-hifenizadas da qual o *Patgen* extrai padrões para criar regras de hifenização. Para garantir a funcionalidade adequada do *Patgen*, o arquivo de tradução e o arquivo de dicionário devem utilizar a mesma codificação, mesmo que seja uma codificação de vários bytes. O arquivo de tradução descreve como lidar com sequências de bytes que representam um glifo, e o *Patgen* funcionará perfeitamente quando houver no máximo 243 símbolos no idioma especificado.

A sintaxe para executar o *patgen* é apresentada na Listagem 2.

```
patgen arquivo_dicionario
   padroes_iniciais arquivo_saida
   arquivo_traducao
```

Listagem 2: Sintaxe para executar o *patgen*.

O arquivo de dicionário é uma lista de palavras hifenizadas corretamente, uma por linha; o arquivo de padrões iniciais é um conjunto de regras de hifenização a ser usado como ponto de partida; o arquivo de saída é o conjunto de regras finais criadas pelo *patgen*; e o arquivo de tradução, que mapeia as várias formas como cada símbolo de um idioma pode aparecer em documentos TEX. Se desejar executar o *patgen* do zero, começando com um conjunto vazio de regras, basta usar um arquivo vazio como o conjunto de padrões iniciais.

O *Patgen* utiliza alguns parâmetros ao longo de sua execução:

hyph_start, hyph_finish Dois números entre 1 e 9 (separados por um espaço) representam os níveis de padrão desejados no conjunto final de regras. Níveis de padrão ímpares são níveis de hifenização e níveis de padrão pares são níveis de inibição. Valores de nível mais alto prevalecem sobre os mais baixos, criando exceções, exceções sobre exceções e assim por diante. `hyph_start` e `hyph_finish` especificam o primeiro e o último nível, respectivamente, a serem considerados durante o processo de criação de regras.

pat_start, pat_finish Os padrões em cada nível são escolhidos em ordem crescente de tamanho (geralmente começando com tamanho 2). Isso é controlado pelos parâmetros `pat_start` e `pat_finish` especificados no início de cada nível. Estes representam o comprimento mínimo e máximo dos padrões de nosso interesse. Seus valores variam entre 1 e 15.

good weight, bad weight, threshold Cada nível de padrões é testado em todas as palavras do dicionário. Um padrão só é aceito se satisfaz a seguinte condição: $\alpha \times \#\text{good matches} - \beta \times \#\text{bad matches} \geq \eta$, onde α é o `good weight` (peso bom), β é o `bad weight` (peso ruim) e η é o `threshold` (limiar).

4. Sistemas de escrita

Existem diferentes sistemas de escrita, categorizados em logográficos, silábicos e alfabéticos. Embora distintos, esses sistemas podem ser construídos a partir da interação entre essas categorias (Coulmas, 2003; Palmer, 2010). Os princípios que orientam um sistema de escrita alfabético ou fonêmico variam significativamente de acordo com a língua e sua história. Aqui estão os principais pontos a serem destacados:

Representação fonêmica O princípio mais comum envolve representar os sons de uma língua por símbolos escritos, usando um

símbolo ou uma combinação de símbolos para representar cada som (por exemplo, sistema ortográfico do português usa um alfabeto de 26 letras e símbolos diacríticos que podem modificar o som de algumas letras ou indicam diferenças de pronúncia e acentuação, como o acento agudo (´), o acento grave[5] (`), o acento circunflexo (ˆ), o til (˜) e a cedilha (ç)).

Etimologia A etimologia também desempenha um papel crucial, já que a grafia de uma palavra reflete sua origem e seu desenvolvimento histórico (por exemplo, a língua francesa muitas vezes reflete as raízes latinas das palavras em sua grafia).

Morfologia A morfologia serve como guia para estruturar muitas línguas, usando letras ou símbolos específicos para indicar terminações, prefixos ou sufixos de palavras (por exemplo, o russo usa diferentes formas do alfabeto cirílico para indicar gênero e caso em seus substantivos).

Evolução da língua A evolução de uma língua ao longo do tempo também contribui para sua forma escrita (por exemplo, a grafia de muitas palavras em inglês mudou ao longo do tempo para refletir as mudanças na pronúncia).

Os sistemas ortográficos de línguas com escrita logográfica, como ideogramas chineses, escrita cuneiforme e hieróglifos egípcios, diferem significativamente dos sistemas de escrita alfabética. Nos sistemas logográficos, símbolos ou caracteres individuais representam palavras ou ideias inteiras, e não fonemas ou sons. Como resultado, os princípios que orientam seus sistemas ortográficos baseiam-se mais em conceitos semânticos e visuais do que em princípios fonêmicos. A hifenização é uma característica das escritas alfabéticas, uma vez que nesses sistemas palavras são formadas a partir de letras. Espaços ou hifens são usados para separar palavras, sílabas ou partes de palavras. Sistemas de escrita logográfica, como o chinês, funcionam de maneira diferente e não utilizam hifenização (Honorof & Feldman, 2006).

4.1. Sistema de escrita do português

O português utiliza um sistema de escrita alfabética, o que significa que sua ortografia é orientada por princípios fonológicos (Cagliari, 2015). A correlação entre escrita e pronúncia influencia a hifenização das palavras, pois elas são divididas em sílabas com base no sistema fonêmico. É importante observar que diferentes idiomas seguem princípios diversos para a divisão de palavras. Por exemplo, o inglês é guiado principalmente por princípios morfológicos, evidentes em palavras como *walk-ing*, *un-happy*, *work-s* e *ear-ly*. Além disso, outros fatores também influenciam a hifenização em inglês, como a distinção entre vogais longas e curtas, que funcionam no contexto de sílabas abertas ou fechadas, respectivamente; e a presença de consoantes duplas e dígrafos (Lin, 2011; Yavas, 2020). Embora cada idioma tenha seus princípios orientadores para o processo de hifenização, múltiplos fatores entram em jogo, levando a várias soluções em certos cenários. Por exemplo, em português, o princípio fonológico levaria a *hipe-rativo*, enquanto o princípio morfológico levaria a *hiper-inflação*. Ambas as abordagens parecem válidas e expressam comportamentos linguísticos distintos, em decorrência de questões semânticas[6]. Algumas palavras raras, como *hiperalgesia*, podem não estar sujeitas a influências morfológicas. Como tem baixa frequência de ocorrência, o indivíduo pode não estar ciente de seus componentes morfológicos e, portanto, hifenizá-la como *hipe-ralgesia*. A forma *hiper-algesia* também poderia ser aceita, enfatizando seus constituintes morfológicos. Além disso, existem inúmeras exceções que podem ser categorizadas em regras específicas.

O simples fato de um sistema ortográfico ser guiado por questões fonológicas não significa necessariamente que suas regras de hifenização espelhem diretamente a contraparte fonética. Isso é particularmente evidente no português, já que não há uma correspondência estrita entre letras e sons. O sistema ortográfico opera de acordo com suas regras próprias. Por exemplo, considere grupos consonantais que criam um único som (dígrafos) em palavras como *a<u>ch</u>ado*, *i<u>lh</u>a*, *su<u>sh</u>i*, *ca<u>rr</u>o* e *ma<u>ss</u>a*. Embora esses dígrafos sejam pronunciados como um único som dentro de uma única sílaba, sua representação na escrita determina como são divididos. Especificamente, consoantes diferentes dentro de um dígrafo devem permanecer juntas, enquanto consoantes idênticas são separadas. Como resultado, observamos hifenizações como *a-cha-do*, *i-lha* e *su-shi*, mas *car-ro* e *mas-sa*.

[5] O acento grave no português não modifica sons. Ele é utilizado para marcar a contração de duas vogais iguais.

[6] Observando palavras com prefixos, parece que mesmo nessas palavras a abordagem fonológica foi predominante na hifenização, mas não podemos dizer se é um subproduto da hifenização automática que pode ser usada para hifenizar palavras em dicionários online.

Em português, o hífen é permitido nos limites das sílabas e, em geral, segue princípios fonológicos. No entanto, de acordo com as gramáticas (Cunha & Cintra, 2016; Bergström & Reis, 2011; Cegalla, 2020), algumas regras específicas ainda podem se aplicar:

Regras de Não-Separação

1. ditongos e tritongos não devem ser separados (por exemplo, *mui-to*, *Pa-ra-guai*);
2. as sequências *ia*, *ie*, *io*, *oa*, *ua*, *ue* e *uo*, quando em posição final átona, não devem ser separadas (por exemplo, *gló-ria*, *vi-tó-ria*, *cá-rie*, *es-pé-cie*, *Má-rio*, *má-goa*, *ré-gua*, *tê-nue*, *con-tí-guo*, *am-bí-guo*);
3. grupos consonantais iniciais de sílaba não devem ser separados (por exemplo, *pneu-má-ti-co*, *psi-có-lo-go*, *mne-mô-ni-co*);
4. os dígrafos *ch*, *lh*, *nh* não devem ser separados (por exemplo, *ra-char*, *a-bro-lhos*, *ma-nhã*);
5. bigramas como *gu* e *qu*, cujo *u* não tem valor fonético, nunca são separados da vogal ou ditongo que os seguem (por exemplo, *U-ru-guai*, *pe-que*);
6. como formam um dígrafo, a vogal e a marca de nasalização subsequente (*m* ou *n*)não devem ser separados (por exemplo, *am-bição*, *man-cha*);
7. ditongos decrescentes não devem ser separados (por exemplo, *ai-ro-so*, *cau-te-la*, *ca-dei-ra*, *cha-péu*, *o-ra-ção*, *noi-te*, *ca-la-bou-ço*, *as-te-rói-de*, *re-tri-bui*);
8. ditongos crescentes não devem ser separados (por exemplo, *his-tó-ria*, *má-goa*, *sé-rio*, *gló-ria*, *fre-quen-te*, *pá-tria*);
9. dissílabos com apenas uma vogal por sílaba não devem ser quebrados (por exemplo, *ato*, *rua*, *ódio*, *unha*);
10. palavras com mais de duas sílabas, ao serem separadas, não podem deixar uma sílaba formada por uma vogal sozinha (por exemplo, *agos-to*, *la-goa*, *ida-de*);

Regras de Separação

11. vogais em hiato e sequências vocálicas em que cada vogal pertence a uma sílaba diferente devem ser separadas (por exemplo, *sa-ú-de*, *ra-i-nha*, *do-er*, *vo-os*), o mesmo procedimento é usado para separar ditongos em sílabas diferentes (por exemplo, *cai-ais*), ou ditongo e vogal em sílabas diferentes (por exemplo, *en-sai-os*);
12. sequências consonantais, quando em sílabas diferentes, devem ser separadas (por exemplo, *af-ta*, *ab-di-car*, *res-ci-são*, *ab-so-lu-to*);
13. os seguintes dígrafos consonantais devem ser separados: *rr*, *ss*, *mm*, *nn*, *sc*, *sç* e *xc* (por exemplo, *ter-ra*, *pro-fes-sor*, *co-mum-men-te*, *con-nos-co*[7], *des-cer*, *cres-ça*, *ex-ce-der*).

Regras 9 e 10 têm como objetivo principal garantir a legibilidade adequada do texto, alinhando-se com a abordagem do TEX para lidar com viúvas e órfãs tipográficas. Como mencionado na Seção 3, as variáveis `lefthyphenmin` e `righthyphenmin` controlam o tamanho mínimo para fragmentos de palavras hifenizadas. Quando essas variáveis são definidas com valores maiores que um, as regras 9 e 10 se tornam regras desnecessárias na hifenização do TEX. No entanto, a hifenização completa das palavras pode ser útil, principalmente em aplicativos de conversão de texto em voz (Libossek & Schiel, 2000; Trogkanis & Elkan, 2010).

Além disso, é aconselhável evitar a separação de dissílabos de quatro letras (por exemplo, *para*, *como*, *cede*). Essas considerações também levam a um texto mais esteticamente agradável e inteligível e ao controle do TEX sobre fragmentos isolados, por meio das variáveis `\lefthyphenmin` e `\righthyphenmin`.

Em algumas situações, o hífen deve ser repetido no início da linha seguinte. São elas:

1. casos em que palavras compostas ligadas por hífen são separadas entre linhas (por exemplo, *couve-/-flor*, *ex-/-presidente*); e
2. casos em que a separação de um pronome possa resultar em alteração de sentido (por exemplo, *prazer de ver-/-me*).

Sistematizar as regras que orientam os limites silábicos e a hifenização do português é um passo fundamental para entender e melhorar as regras de hifenização do TEX. Para garantir a precisão das regras de hifenização e comparar seus resultados de forma eficaz, um dicionário ortográfico de hifenização funciona como uma referência indispensável. Esse dicionário fornece uma lista abrangente de palavras hifenizadas corretamente, permitindo o cálculo da exatidão de cada conjunto de regras. Ao consultar o dicionário ortográfico de hifenização, é possível verificar se

[7] É importante ressaltar que as formas *connosco* e *commummente* não são usadas em todas as variedades do português. Por exemplo, os europeus adotam essas formas, enquanto os brasileiros não.

os padrões de hifenização gerados por um conjunto de regras estão de acordo com os padrões estabelecidos. Esse processo envolve analisar o quão bem as regras de hifenização aderem às convenções aceitas do idioma, garantindo que elas segmentem palavras de forma eficaz, sem comprometer a legibilidade ou a consistência. Além disso, comparando os resultados obtidos de diferentes conjuntos de regras de hifenização com as entradas do dicionário ortográfico, podemos avaliar a eficácia e a confiabilidade de cada abordagem.

Na Seção 5, definiremos as especificações desse dicionário de referência, delineando seu papel em atualizações e geração de regras subsequentes.

5. Criando o dicionário de hifenização para referência

Nosso conjunto de palavras em português foi criado a partir do corpus CETENFolha (Linguateca, 2014), ampliado com a lista de palavras do Palavras NET (sd.) e com o dump da Wikipédia em português (Wikipedia, 2023). A partir dos dados da Wikipédia, reduzimos a seleção a um subconjunto de 50.721 palavras, que representam 95% das ocorrências no dump (o conjunto inicial possuía 419.578 palavras). Esse limite foi estabelecido para filtrar erros de digitação e palavras pouco frequentes. Por fim, refinamos a lista, mantendo apenas as palavras para as quais encontramos dados de hifenização em pelo menos um dos seguintes recursos lexicais online: Michaelis (sd.), Priberam (sd.), Wikcionário (sd.), Aulete (sd.), Portal da Língua Portuguesa (sd.) e Dicio (sd.). Esse processo de curadoria resultou em um dicionário final contendo 86.324 palavras. Também realizamos correções manuais em 10 entradas que julgamos necessárias.

O fato de a língua portuguesa ter a fonologia como fator-chave para a hifenização exige uma compreensão de como a acentuação influencia esse processo. Toda palavra no idioma com duas ou mais sílabas possui uma sílaba mais proeminente, considerada sílaba tônica[8]. A tonicidade pode recair sobre qualquer uma das três últimas sílabas da palavra, contadas a partir da margem direita. As palavras são classificadas como oxítonas quando a última sílaba é tônica, paroxítonas quando a penúltima sílaba é tônica, ou proparoxítonas quando a antepenúltima sílaba é tônica. O padrão de tonicidade considerado não marcado (o mais comum) não recebe marcação diacrítica; os padrões marcados, incluindo todas as proparoxítonas, recebem uma marcação diacrítica. Essa marcação serve para indicar ao leitor que a sílaba tônica segue um padrão menos comum (marcado). O uso de marcas diacríticas para indicar a tonicidade é uma pista importante para determinar como uma palavra pode ser hifenizada. Por exemplo, as palavras *saúde* e *saída*, que possuem a tonicidade marcada nas vogais *u* e *i*, respectivamente, indicam claramente que essas vogais formam sílabas próprias: *sa-ú-de* e *sa-í-da*. No entanto, existem casos ambíguos, tanto para falantes, quanto na literatura linguística, como *his-tó-ria* (ou *his-tó-ri-a*) e *car-tó-rio* (ou *car-tó-ri-o*). A questão é se grupos vocálicos finais como *-ia* e *-io* são ditongos ou hiatos; decisão essa que mudaria tanto a hifenização da palavra como a posição em que a sílaba tônica se encontra. Adotamos, nesta pesquisa, a posição de autores como Cunha & Cintra (2016); Cegalla (2020); Pestana (2022), os quais assumem que palavras que terminam em ditongos crescentes recebem acento gráfico e são, portanto, paroxítonas, embora se reconheça que podem ser também proparoxítonas aparentes Cunha & Cintra (2016); Cegalla (2020).

No Wikcionário português, encontramos 1.848 palavras marcadas como proparoxítonas aparentes. As proparoxítonas aparentes são palavras que parecem ter a sílaba tônica na antepenúltima sílaba devido a sequências vocálicas pós-tônicas tratadas como ditongos crescentes. No entanto, para serem verdadeiras proparoxítonas, essas sequências devem formar um hiato; se forem ditongos, as palavras são classificadas como paroxítonas. Apesar da aparência, elas seguem as regras de acentuação das paroxítonas. As proparoxítonas aparentes podem gerar confusão na silabação e na hifenização porque seu padrão de tonicidade sugere diferentes pontos de quebra. Por exemplo, a palavra *início* poderia ser hifenizada de duas maneiras: *i-ní-ci-o* (proparoxítona) ou *i-ní-cio* (paroxítona). Essa duplicidade surge do tratamento das sequências vocálicas finais como sílabas separadas ou como partes de ditongos. Do ponto de vista linguístico, a silabificação acertada deve considerar a pronúncia pretendida e a estrutura morfológica, alinhando-se com as regras de acentuação padrão do idioma (Senado Federal, 2013).

A fim de analisarmos o desempenho das regras de hifenização e de propormos melhorias, extraímos quatro sub dicionários do dicionário principal: um primeiro dicionário em que todos dicionários online compartilham uma mesma

[8] É preciso destacar, entretanto, que há na língua portuguesa monossílabos tônicos, como *pá*, *pé* e *pó*. Não a incluímos, tendo em vista que elas não estão dentro do escopo da presente pesquisa.

hifenização (dicionário 6, com 15.842 palavras); um segundo dicionários em que cinco dicionários online propõem uma mesma hifenização (dicionário 5, com 15.642 palavras); um terceiro dicionário em que quatro dicionários online utilizam uma mesma hifenização (dicionário 4, com 10.299 palavras); e, por fim, um quarto dicionário em que três dicionário online propõem uma mesma hifenização (dicionário 3, com 7.745 palavras).

Em caso de desacordo entre diferentes hifenizações propostas pelos dicionários online, devemos ter cautela e examinar cada caso individualmente. Poderíamos ficar tentados a escolher a abordagem de maioria simples, selecionando apenas as hifenizações que reúnam mais defensores. No entanto, não está claro como as hifenizações nesses dicionários online foram curadas; muitos podem usar abordagens algorítmicas, levando a possíveis falhas compartilhadas entre eles. Por exemplo, cinco dicionários usam a hifenização *quart-zo*, enquanto o dicionário Aulete (originário de Portugal) favorece a hifenização *quar-tzo*, o que pode ser explicado pela ausência de vogal epentética no português europeu[9], conforme discutido em Collischonn (1999), que menciona Mateus (sd.). Similarmente, *ap-nei-a* ou *a-pnei-a* e *disp-nei-a* ou *dis-pnei-a* devem ser aceitas considerando que pode haver uma vogal epentética (português brasileiro) ou não (português europeu). Justificativa análoga pode ser atribuída a *hiperalgesia*, já mencionada na Seção 4.1. Dentre outros exemplos, podemos destacar também *neerlandês*, que aparece hifenizado como *ne-er-lan-dês* em cinco dicionários e como *neer-lan-dês* em um. Se considerarmos a pronúncia, preferiríamos a primeira hifenização, mas a segunda se conforma melhor às sílabas escritas regulares em português. Algumas sequências podem aceitar mais de uma hifenização, pois representam palavras diferentes. Por exemplo, *sub-li-nha* e *su-bli-nha*, onde a primeira é o substantivo *sublinha* e a segunda é uma forma flexionada do verbo *sublinhar*.

As proparoxítonas aparentes são casos de ambiguidade em que alguns dicionários optam por um hiato final (proparoxítonas), enquanto outros optam pelo ditongo final (paroxítonas) e alguns preferem marcar a possibilidade de existir ou não a separação entre as duas vogais finais. Para lidar com esses casos[10], utilizamos o seguinte procedimento: a) se a palavra tiver acento gráfico, selecionamos a opção paroxítona (por exemplo, *po-lí-cia* em vez de *po-lí-ci-a* e *pe-rí-neo* em vez de *pe-rí-ne-o*); b) se não tiver acento gráfico, escolhemos a opção proparoxítona (por exemplo, *au-to-ri-a* em vez de *au-to-ria* e *sim-bo-lo-gi-a* em vez de *sim-bo-lo-gia*)[11].

6. Regras de hifenização para o português

de Rezende (1987) criou os padrões iniciais para hifenização do português no TEX. O conjunto de padrões foi atualizado pela última vez em 2015, incorporando contribuições de de Rezende & Almeida (2015), resultando em um total de 307 regras.[12] Essas regras hifenizam efetivamente a maioria das palavras portuguesas.

Dentre as regras padrão do TEX, 252 (82%) seguem o padrão `1CV`, que representa as sílabas CV recorrentes em português. No que diz respeito às consoantes escritas, existem tipicamente 18 consideradas (`b`, `c`, `ç`, `d`, `f`, `g`, `j`, `k`, `l`, `m`, `n`, `p`, `r`, `s`, `t`, `v`, `x`, e `z`) que podem ser combinadas com 14 vogais escritas (`a`, `e`, `i`, `o`, `u`, `á`, `â`, `ã`, `é`, `í`, `ó`, `ú`, `ê`, e `õ`). Essas combinações resultam em padrões CV que indicam pontos de hifenização favoráveis antes das consoantes[13]. Vale ressaltar que pode haver outras regras ou exceções nos padrões de hifenização não cobertas por essas regras padrão. Para acomodar exceções, as seguintes regras foram propostas por de Rezende (1987):

- 20 regras foram criadas para casos envolvendo consoantes `b`, `c`, `d`, `f`, `g`, `k`, `p`, `t`, `v` ou `w`[14] seguidas de `l` ou `r`;

[9] O dicionário online Aulete tem origem na versão impressa conhecida como *Dicionário Caldas Aulete*. É um dicionário de Portugal, refletindo a variedade do português europeu.

[10] Em termos de translineação, devemos aplicar regras que impeçam uma vogal isolada na linha seguinte, entretanto, como mencionamos anteriormente, o TEX já faz este controle. O que queremos aqui é escolher uma forma de separação silábica que iremos adotar para ser base para as regras de hifenização.

[11] Apenas precisamos ter certeza de não selecionar aqueles casos em que há uma sequência da forma `[gq]u-[aeio]` (usando notação regex), como em *a-pa-zi-gu-ar* e *en-xa-gu-ar*.

[12] O conjunto de regras para hifenização do português é pequeno em comparação com regras de outros idiomas. Por exemplo, o inglês atualmente possui 4.938 regras, o russo possui 7.023 regras e o alemão possui 34.011 regras.

[13] Em português, geralmente a letra *h* aparece na posição inicial da palavra ou antes das consoantes *c*, *l* ou *n*. Portanto, não é interessante incentivar um ponto de hifenização antes de um *h*V. Exemplos de palavras que possuem um *h* no meio e não são precedidas por *c*, *l* ou *n* são: *Bahia*, *Corinthians*, *show*, *shopping*, *Matheus* e *sushi*. Esses são nomes próprios ou palavras estrangeiras cuja grafia não foi adaptada ao português.

[14] Não se sabe ao certo por que adicionaram as regras `1w2l` e `1w2r`, já que não há palavras no português com essa sequência. Encontramos apenas palavras estrangeiras como *showroom*, *wrestling*, *bowling* ou outras palavras estrangeiras bem raras.

- 3 regras para `c`, `l` ou `n` seguidas de `h`;
- 23 padrões distintos foram introduzidos para indicar pontos de hifenização entre vogais ou entre `c`'s, `r`'s e `s`'s;
- 8 padrões seguem o padrão `1[gq]u4V`, sinalizando pontos de hifenização favoráveis antes de `g` ou `q`, seguidos por uma sequência de `u` e uma vogal, com um ponto de inibição entre o `u` e a vogal subsequente;
- 1 padrão representado como `1-`, indicando que um hífen realmente funciona como um ponto de hifenização favorável.

No entanto, diante de certos casos, uma análise das regras padrão foi realizada para identificar áreas para melhoria (Araujo & Benevides, 2024). Ao lidar com problemas específicos e considerar padrões atípicos, nosso objetivo foi aumentar a precisão da hifenização. A metodologia utilizada para esta análise será descrita a seguir.

6.1. Atualizações para as regras de hifenização do português

O primeiro passo para propor atualizações ao conjunto de regras de hifenização é avaliar o desempenho delas em nosso dicionário de hifenização de referência. A Tabela 1 resume os resultados, mostrando o número de palavras corretamente hifenizadas pelas regras padrão, o número de palavras hifenizadas incorretamente e o número de palavras em que um ponto de hifenização foi omitido.

dicionário	correto	incorreto	faltante	palavras
dic. 6	15537	30	277	15842
dic. 5	13981	1146	601	15642
dic. 4	9001	883	518	10299
total	38519	2059	1396	41783

A soma das hifenizações corretas com as incorretas e as faltantes pode ser maior que o número total de entradas, pois alguns casos podem conter hifenizações incorretas e faltantes ao mesmo tempo.

Tabela 1: Resultados da aplicação das regras padrão de hifenização do TEX aos dicionários 6, 5 e 4.

Para criar regras complementares que resolvam os casos de hifenizações incorretas e faltantes, adotamos a seguinte abordagem: começamos resolvendo os casos de hifenizações incorretas e possíveis exceções necessárias para as regras propostas; em seguida, buscamos resolver os casos de hifenizações faltantes, também analisando a necessidade de criarmos regras de exceção. Este processo foi realizado interativamente, começando pelo dicionário 6, passando ao dicionário 5 e terminando no dicionário 4. O dicionário 3 foi reservado para avaliar a generalização das regras propostas.

Abaixo, sistematizamos o conjunto de regras adicionais propostas e fornecemos alguns exemplos de aplicação para cada uma delas. Este conjunto de regras deve ser utilizado de forma adicional às regras propostas por de Rezende & Almeida (2015).

1 `.g2no`, `.g2nó`, `.g2nô` — *gnomo*, *gnóstico*, *gnômon*

2 `t2c` — *tchau*, *tcheco*

3 `1p2neu` — *pneumonia*, *pneumotórax*, *pneumático*, *hidropneumático*

4 `.t2m` — *tmese*

5 `.p2t` — *ptose*, *pterossauro*

6 `.m2n` — *mnemônico*

7 `c2za` — *czar*

8 `.s2` — *stalinismo*

9 `.t2` — *tsunami*, *tzarista*

10 `.p2si`, `.p2sí` — *psicologia*, *psíquico*

11 `su2b3r`, `su2b3l` — *subrotina*, *sublunar*
exceção: `.su3b4li` — *sublinhar*, *sublimar*

12 `.ne4o` — *neoliberal*, *neonazista*

13 `1p2seu1d` — *pseudônimo*

14 `1qu` — *enquanto*, *inquieto*, *farquhar*, *quiloqubit*

15 `a1ir.`, `u1ir.` — *sair*, *extrair*, *diminuir*, *incluir*

16 `a1ind`, `a1i1nh` — *ainda*, *rainha*

17 `e1imp` — *reimpresso*, *teleimpressor*

18 `e1inc`, `e1inf`, `e1ing`, `e1ins`, `e1int`, `e1inv` — *reincidência*, *reinfecção*, *reingressa*, *reinserção*, *reintegração*, *reinventar*

19 `u1iz.`, `a1iz.` — *juiz*, *raiz*

20 `pro1i1b` — *proibição*

21 `tu1i`, `bu1i`, `nu1i`, `o1im`, `o1in`, `u1in`, `su1i`, `í1e`, `ju1i`, `fu1i`, `du1i`, `do1im`, `au1i`, `u1i1ç` — *intuitivo*, *contribuidor*, *ingenuidade*, *coimbra*, *coincide*, *ruindade*, *suicida*, *píer*, *juizado*, *fuinha*, *assiduidade*, *amendoim*, *cacauicultor*, *constituição*
exceção: `tu2id`, `tu2it`, `co2ima`, `o2i1na` — *gratuidade*, *intuito*, *coima*, *boina*

22 `a1â`, `a1ã`, `a1é`, `a1í`, `a1ó`, `a1ô`, `a1ú`, `e1á`, `e1â`, `e1ã`, `e1é`, `e1ê`, `e1í`, `e1ó`, `e1ô`, `e1ú`, `é1o`, `i1á`, `i1ã`, `i1é`, `i1í`, `i1ó`, `i1u`, `i1ú`, `í1a`, `í1o`, `o1á`, `o1ã`, `o1é`, `o1ê`, `o1í`, `o1ó`, `u1á`, `u1ã`, `u1â`, `u1é`, `u1ê`, `u1í`, `ú1o` — *abraâmico*, *abraão*, *aéreo*, *país*, *caótico*, *faraônico*, *saúde*, *balneário*, *oceânico*, *campeã*, *feérico*, *preênsil*, *veículo*, *teórico*, *napoleônico*, *conteúdo*, *néon*, *diário*, *região*, *soviético*, *iídiche*, *periódico*, *feiura*, *viúva*, *maníaco*, *íon*, *razoável*, *joão*, *poético*, *boêmia*, *heroísmo*, *alcoólico*, *usuário*, *itapuã*, *lituânia*, *suécia*, *cauê*, *suíça*, *flúor*
exceção: `1gu2á`, `1gu2ã`, `1gu2é`, `1gu2ê`, `1gu2í`, `1qu2á`, `1qu2ã`, `1qu2â`, `1qu2é`, `1qu2ê`, `1qu2í`, — *jaraguá*, *saguão*, *alguém*, *português*, *linguística*, *aquático*, *camaquã*, *equânime*, *inquérito*, *sequência*, *química*

23 `1bô`, `1cô`, `1çô`, `1dô`, `1fô`, `1gô`, `1lô`, `1mô`, `1nô`, `1pô`, `1rô`, `1sô`, `1tô`, `1vô`, `1xô`, `1zô` — *robô*, *recôncavo*, *maçônico*, *judô*, *telefônica*, *xangô*, *camelô*, *capô*, *sumô*, *econômico*, *tarô*, *subsônico*, *chatô*, *vovô*, *saxônia*, *amazônia*

24 `4a.`, `4e.`, `4o.` — *secretária*, *planície*, *paratormônio*

O resultado da hifenização do novo conjunto de regras é apresentado na Tabela 2. Observamos uma redução significativa do número de hifenizações incorretas, entretanto, há aumento no número de casos em que pontos de hifenização não foram identificados. Isto ocorre devido às regras `4a.`, `4e.` e `4o.` (sugeridas em 24).

Ao analisarmos o dicionário 5, o número de hifenizações incorretas aumentou substancialmente. A maioria desses erros decorreu de proparoxítonas aparentes. Esses casos foram facilmente corrigidos evitando o hífen final que deixa as vogais *a*, *e* ou *o* sozinhas na última sílaba. Para tanto, foram introduzidas essas regras em 24. Embora um conjunto de regras mais abrangente pudesse ser desenvolvido para levar em conta todos os cenários, optamos por regras mais concisas, apesar de suas limitações. A adição delas reduz o número de hifenizações incorretas; no entanto, aumenta o número de hifenizações ausentes. A grande maioria dessas hifenizações ausentes adicionais ocorre antes da vogal final de uma palavra. Considerando o menor número de casos e a posição final (antes de uma vogal final), a gravidade percebida (uma hifenização ausente é considerada menos problemática do que uma incorreta, sobretudo em final de palavra) é insignificante; portanto, optamos por mantê-las.

dicionário	correto	incorreto	faltante	palavras
dic. 6	15304	0	538	15842
dic. 5	14814	20	818	15642
dic. 4	9690	13	603	10299
total	39808	33	1959	41783

A soma das hifenizações corretas com as incorretas e as faltantes pode ser maior que o número total de entradas, pois alguns casos podem conter hifenizações incorretas e faltantes ao mesmo tempo.

Tabela 2: Resultados da aplicação dos novo conjunto de regras na hifenização das palavras dos dicionários 6, 5 e 4.

7. Elaboração de novas regras usando o *patgen*

Em Araujo & Benevides (2024), também abordamos a criação de novas regras utilizando o *patgen*. Este processo envolve a definição de parâmetros específicos, a escolha de um dicionário de referência com hifenizações corretas e a utilização de um arquivo de tradução personalizado para o *patgen*. Na Seção 3.1, detalhamos como o *patgen* funciona, quando mencionamos a necessidade de utilizarmos um arquivo de tradução para a língua em questão. As primeiras 10 linhas do arquivo de tradução utilizado para o português é apresentado na Listagem 3.

Listagem 3: 10 primeiras linhas do arquivo de tradução para o português.

```
 1 1
%% This file portuguese.tra defines the
    letters used for generating
%% Portuguese hyphenation patterns with
    patgen.
 a A
 á Á
 à À
 â Â
 ã Ã
 b B
 c C
```

Também mencionamos anteriormente que o *patgen* necessita de um dicionário de hifenização de referência, que será sua base de dados para extrair padrões. O dicionário de referência utilizado foi a junção dos dicionários 6, 5 e 4, sendo que o 3 foi reservado para testar o conjunto final de regras elaboradas.

Ainda, como destacado anteriormente na Seção 3.1, precisamos definir certos parâmetros numéricos na execução do *patgen*. Existem infinitas maneiras de especificar tais parâmetros. Com base em trabalhos anteriores, optamos por testar duas abordagens: (1) utilização de parâmetros fixos em todas as etapas da execução do *patgen* (partindo de um conjunto vazio ou das regras

padrão) (Haralambous, 2021); e (2) utilização de uma abordagem progressiva, começando com um conjunto vazio e criando regras de ordem superior a cada etapa, sempre com base nas regras da etapa anterior (Sojka & Sojka, 2019).

Embora os resultados do conjunto de regras geradas pelo *patgen* pareçam excepcionais no conjunto de palavras usadas para criá-las, devemos ser céticos em relação à sua generalização. Para investigar esse assunto, utilizamos essas regras no dicionário 3. Como demonstrado em Araujo & Benevides (2024), o *patgen* cria um grande conjunto de regras, muitas delas longas e específicas, resultando em baixa capacidade de generalização e desempenho inferior às regras manuais propostas no dicionário 3. Resultados semelhantes foram observados quando o *patgen* começa a partir do conjunto padrão de regras ou do nosso conjunto de regras corrigidas.

word	hyphenation	dictionary					
		Michaelis	Priberam	Wikitionary	Aulete	Portal	Dicio
sublinhar	su-bli-nhar		x	x	x		
	sub-li-nhar	x				x	
reiniciar	rei-ni-ci-ar			x		x	
	re-i-ni-ci-ar	x	x		x		x
ciberespaço	ci-be-res-pa-ço		x			x	
	ci-ber-es-pa-ço	x		x	x		x
hiperalgesia	hi-pe-ral-ge-si-a	x	x		x	x	x
	hi-per-al-ge-si-a			x			
autoimagem	au-toi-ma-gem					x	
	au-to-i-ma-gem	x			x		x

Tabela 3: Hifenização de palavras evidenciando o conflito entre o critério morfológico e o critério fonológico na hifenização utilizada nos dicionários.

8. Limitações

De forma geral, embora as palavras incorretas compartilhem algumas características comuns que poderiam permitir, até certo ponto, a redução de erros de hifenização, existe uma limitação inerente à maneira como as regras do TEX são concebidas. Abaixo, apresentamos os padrões encontrados nesses dados que poderiam ser contemplados por uma estrutura de regras diferente.

Determinação Morfológica Prefixos como *re-*, *sub-*, *ciber-*, *hiper-* e *auto-*, entre outros, exigem a separação do prefixo de seu radical, o que pode levar a questões fonológicas. Por exemplo, palavras como *reiniciar*, *sublinhar*, *ciberespaço*, *hiperalgesia* e *autoimagem* contêm prefixos e poderiam ser hifenizadas como *re-i-ni-ci-ar*, *sub-li-nhar*, *ci-ber-es-pa-ço*, *hi-per-al-ge-si-a* e *au-to-i-ma-gem*, respectivamente, para respeitar sua formação morfológica. No entanto, considerando que o português hifeniza suas palavras com base em correlatos fonológicos silábicos, palavras que requerem informação morfológica, como essas, podem não ter sua hifenização realizada corretamente. Estes exemplos são apresentados na Tabela 3, comparando-se a forma como diferentes dicionários as hifenizam.

Estrangeirismos Um grupo de palavras do corpus é formado por terminologias ou vocábulos incorporados ao português sem completa adaptação fonológica, como *darwinismo*, *quilowatt* e *esfiha*. Essa falta de adaptação resulta em padrões fonológicos altamente específicos para cada palavra, tornando impraticável integrá-las às regras do TEX ao lado de outras. Podemos, entretanto, adicioná-las à lista de exceções de hifenização. Para palavras como *farquhar* e *qubit*, que são estrangeiras, mas, apesar de raras, possuem frequência significativa, adicionamos a regra 14, já que tal regra de hifenização é coerente com as normas da língua portuguesa e não há necessidade de criar qualquer regra de exceção.

Grupos consonantais iniciais O português possui poucos casos de grupos consonantais no início de uma palavra. Geralmente, são resquícios etimológicos e, atualmente, improdutivos na língua, já que não há neologismos com esse padrão. Encontros como *ps-* e *pn-* são mais frequentes, pois estão presentes em palavras como *psicologia* e *pneu*, de média frequência na língua. Esses casos podem ser previstos por regras específicas que cobririam 49 e 13 palavras no corpus, respectivamente. No entanto, existem grupos consonantais encontrados em palavras bem específicas e de baixa frequência. Embora possível, não vale a pena adicionar regras muito específicas para grupos encontrados em palavras como *dzeta*, *gnu*, *cnidário*, *ftálico* e *gnaisse* – que representam apenas cinco palavras.

Abreviações, Acrônimos e Siglas Seja por eficiência, conveniência, clareza ou jargão especializado, é comum usar versões abreviadas de palavras ou frases. A abreviação

é um método empregado para alcançar encurtamento. Em nosso corpus de português, encontramos exemplos como *etc.*, *Dr.*, *Exmo.*, *cap.*, *Univ.*, *ed.*, *s.n.*. Outra forma abreviada é a sigla, que consiste em usar as letras iniciais das palavras para criar uma versão reduzida. No entanto, as siglas nem sempre seguem as regras de hifenização descritas neste trabalho, pois não necessariamente seguem os padrões ortográficos ou fonotáticos da língua. Algumas siglas encontradas no corpus incluem *SESC*, *INSS*, *PCdoB*, *PM* e *UFRJ*. Os acrônimos são um tipo específico de abreviação em que as primeiras letras (ou grupos de letras) de cada palavra são combinadas para formar uma nova palavra pronunciável. No corpus, encontramos exemplos como *Anatel*, *Ovni*, *Sida* e *Mercosul*. Essas várias formas abreviadas desempenham um papel importante na linguagem escrita, fornecendo maneiras concisas de representar palavras ou frases mais longas. É importante observar que abreviações, acrônimos e siglas geralmente são tratados como unidades únicas e não são hifenizados.

9. Conclusão

Neste estudo, exploramos as regras padrão de hifenização do TEX e analisamos as regras adicionais para melhorar o desempenho da hifenização em português. Ao abordar as limitações das regras antigas e das abordagens existentes no TEX, e ao incorporar considerações morfológicas e fonológicas, conseguimos reduzir significativamente o número de erros de hifenização. Contudo, o conjunto de regras adicional ainda não é suficiente para alcançar uma precisão perfeita, o que nem mesmo é desejado, pois nosso conjunto de dados pode conter ruído e há muitos casos de hifenização duvidosa. Optamos por um conjunto conciso de regras que possa generalizar melhor e, portanto, se alinhar adequadamente com as regras subjacentes de hifenização do idioma.

Também testamos regras criadas pelo *patgen*, que geraram um extenso conjunto de regras incapazes de generalizar de forma eficaz. Em contraste, nossas regras manuais apresentaram desempenho superior em um conjunto de validação.

Diante disso, embora nosso conjunto de regras aprimorado demonstre melhorias significativas na precisão da hifenização, ainda há espaço para refinamentos adicionais. Aspectos como o suporte a caracteres especiais e a segmentação silábica universal, conforme considerado por Sojka et al. (2023), merecem consideração. A incorporação de expressões regulares, a criação de classes de caracteres e o suporte à posição de acento tônico das palavras podem aumentar ainda mais a eficiência e a generalidade do sistema de hifenização. Tais aprimoramentos não apenas aumentariam a precisão, mas também tornariam o sistema mais robusto e adaptável às diversas nuances do português.

Referências

de Araújo, Antonio Martins & Toru Maruyama. 2015. A hifenização em português. *Idioma* 28. 90–107.

Araujo, Leonardo & Aline Benevides. 2024. Enhancing TEX hyphenation rules for Portuguese. *TUGboat* 45(3). 309–316. 10.47397/tb/45-3/tb141araujo-pthyph

Aulete. sd. Dicionário Aulete digital. Accessed: 2023-06-26.

Bergström, Magnus & Neves Reis. 2011. *Prontuário ortográfico e guia da língua portuguesa*. Casa das Letras

Cagliari, Luiz Carlos. 2015. Aspectos teóricos da ortografia. Em Maurício Silva (ed.), *Ortografia da língua portuguesa: história, discurso, representações*, 17–52. Contexto

Cegalla, Domingos Paschoal. 2020. *Novíssima gramática da línngua portuguesa*. São Paulo: Companhia Editora Nacional

Collischonn, Gisela. 1999. A epêntese e a fonologia lexical do português brasileiro. Em *14th Encontro Nacional da Associação Portuguesa de Linguística*, 369–382.

Coulmas, Florian. 2003. *Writing systems: An introduction to their linguistic analysis*. Cambridge University Press

Cunha, Celso & Lindley Cintra. 2016. *Nova gramática do português contemporâneo*. Lexikon Editora Digital 7th edn.

Dicio. sd. Dicionário online de Português. Accessed: 2023-06-26.

Gândavo, Pêro de Magalhães. 1981. *Regras que ensinam a maneira de escrever e a ortografia da língua Portuguesa*. Lisboa: Biblioteca Nacional.

Haralambous, Yannis. 2021. A revisited small tutorial on Patgen, 28 years after. Relatório técnico. CTAN.

Honorof, Douglas & Laurie Feldman. 2006. The chinese character in psycholinguistic research: Form, structure, and the reader. Em Ping Li, Li Hai Tan, Elizabeth Bates & Ovid J. L. Tzeng (eds.), *The Handbook of East Asian Psycholinguistics*, vol. 1, chap. 17, 195–208. Cambridge University Press. 10.2277/0521833337

Hunspell's Team. 2023. Hunspell. [Visitado a 2023-04-13].

Hunspell's Team. sd. Hunspell Hyphen.

Knuth, Donald Ervin. 1977. Preliminary preliminary description of TEX. Online.

Levien, Raph. 1998. *Brief explanation of the hyphenation algorithm herein*. Hunspell.

Liang, Frank & Peter Breitenlohner. 1991. PATtern GENeration program for the TEX82 hyphenator. Relatório Técnico. 2 CTAN

Liang, Franklin Mark. 1983. *Word hy-phen-a-tion by com-put-er*: Stanford University. Tese de Doutoramento

Libossek, Marion & Florian Schiel. 2000. Syllable-based text-to-phoneme conversion for German. Em *6th International Conference on Spoken Language Processing (ICSLP)*, 283–286.

Lin, Li-chin. 2011. Fundamental generalizations of English syllabification. *Concentric: Studies in Linguistics* 37(2). 179–208.

Linguateca. 2014. CETENFolha corpus. Accessed: 2023-06-26.

Mateus, Maria Helena Mira. sd. Questões fonológicas do português. Manuscrito

Michaelis. sd. Michaelis Dicionário brasileiro da língua Portuguesa. Accessed: 2023-06-26.

Németh, László. 2006. Automatic non-standard hyphenation in OpenOffice.org. *TUGboat* 27(1). 32–37

Palavras NET. sd. Palavras NET word list. Accessed: 2023-06-26.

Palmer, David D. 2010. Tokenisation and sentence segmentation. Em Nitin Indurkhya & Fred J. Damerau (eds.), *Handbook of Natural Language Processing*, chap. 2, 9–30. CRC Press

TEX pattern authors. sd. TEX hyphenation patterns.

Pestana, Fernando. 2022. *A gramática para concursos públicos*. Método

Portal da Língua Portuguesa. sd. Portal da língua Portuguesa online dictionary. Accessed: 2023-06-26.

Priberam. sd. Dicionário priberam da língua Portuguesa. Accessed: 2023-06-26.

de Rezende, Pedro Jussieu. 1987. Portuguese hyphenation table for TEX. *TUGboat* 8(2). 102–102

de Rezende, Pedro Jussieu & José Joao Dias Almeida. 2015. Hyphenation patterns for Portuguese.

Scannell, Kevin. 2003. Hyphenation patterns for minority languages. *TUGboat* 24(2). 236–239

Senado Federal. 2013. *Acordo ortográfico da língua portuguesa: atos internacionais e normas correlatas*. Senado Federal

Sojka, Ondřej, Petr Sojka & Jakub Máca. 2023. A roadmap for universal syllabic segmentation. *TUGboat* 44(2). 289–296. 10.47397/tb/44-2/tb137sojka-syllabic

Sojka, Petr. 1995a. Notes on compound word hyphenation in TEX. *TUGboat* 16(3). 290–297

Sojka, Petr. 1995b. Notes on compound word hyphenation in TEX. Relatório técnico. Masaryk University in Brno, Faculty of Informatics

Sojka, Petr. 2002. Hyphenation – a tutorial for TEX users.

Sojka, Petr & David Antoš. 2003. Context sensitive pattern based segmentation: A Thai challenge. Em *EACL Workshop on Computational Linguistics for South Asian Languages–Expanding Synergies with Europe, Budapest*, 65–72

Sojka, Petr & Ondřej Sojka. 2019. The unreasonable effectiveness of pattern generation. *TUGboat* 40(2). 187–193.

Sojka, Petr et al. 2005. *Competing patterns in language engineering and computer typesetting*: Masaryk University, Brno. Tese de Doutoramento

Trogkanis, Nikolaos & Charles Elkan. 2010. Conditional random fields for word hyphenation. Em *48th Meeting of the Association for Computational Linguistics (ACL)*, 366–374.

Wikcionário. sd. Wikcionário online dictionary. Accessed: 2023-06-26.

Wikipedia. 2023. Portuguese wikipedia dump. Accessed: 2023-06-26.

Yavas, Mehmet. 2020. *Applied English phonology*. John Wiley & Sons

PROPOR 2024
Artigos Convidados

Proposta recebida em Junho 2024 e aceite para publicação em Outubro 2024.

Explorando a Eficácia dos Modelos Generativos em Tarefas de Análise de Sentimentos no Português Brasileiro

Exploring the Effectiveness of Generative Models in Sentiment Analysis Tasks in Brazilian Portuguese

Gladson de Araújo
Universidade do Estado do Amazonas

Tiago de Melo
Universidade do Estado do Amazonas

Carlos Maurício S. Figueiredo
Universidade do Estado do Amazonas

Resumo

Modelos de linguagem de grande escala (LLMs) têm se destacado em diversas tarefas de processamento de linguagem natural (PLN). Este artigo investiga sua eficácia em tarefas de análise de sentimentos no contexto do português brasileiro, explorando a identificação de frases opinativas, polaridade e frases comparativas. O estudo avalia o desempenho de modelos como ChatGPT e Sabiá em diferentes tarefas e conjuntos de dados, comparando-os com métodos da literatura. Ainda, exploramos o uso de LLMs na anotação automática de dados. Os resultados demonstram o potencial dos LLMs na análise de sentimentos, especialmente na identificação de polaridade, e discutem suas limitações e aplicações em tarefas de anotação de dados.

Palavras chave

modelos de linguagem de grande escala, análise de sentimentos, anotação automática de dados

Abstract

Large language models (LLMs) have been successfully applied in various natural language processing (NLP) tasks. This paper investigates their effectiveness in sentiment analysis tasks in the context of Brazilian Portuguese, exploring the identification of opinionated sentences, polarity, and comparative sentences. The study evaluates the performance of models such as ChatGPT and Sabiá on different tasks and datasets, comparing them with methods from the literature. Furthermore, we explore the use of LLMs in automatic data annotation. The results demonstrate the potential of LLMs in sentiment analysis, especially in polarity identification, and discuss their limitations and applications in data annotation tasks.

Keywords

large language models, sentiment analysis, automatic data annotation

1. Introdução

Modelos de linguagem de grande escala (LLMs) demonstraram sua capacidade de lidar com uma variedade de tarefas de processamento de linguagem natural (PLN) sem a necessidade de dados de treinamento específicos, um fenômeno denominado aprendizado auto-supervisionado (*self-learning*). Isso é alcançado submetendo o modelo com prompts adequados (Brown et al., 2020). A capacidade de realizar novas tarefas por instrução representa um grande avanço em direção à inteligência artificial para múltiplos propósitos. Embora os LLMs contemporâneos exibam desempenho louvável em certos cenários, elas permanecem propensas a erros no aprendizado *zero-shot* (Chang et al., 2023) e *few-shot* (Li et al., 2023). Além disso, várias configurações, como as definições de temperatura, podem influenciar profundamente a eficácia do modelo. Essas limitações implicam que os LLMs atuais podem não servir verdadeiramente como sistemas de linguagem abrangentes.

O lançamento recente do ChatGPT pela OpenAI atraiu atenção significativa da comunidade de PLN. O ChatGPT[1] é um modelo baseado em Redes Neurais Transformer (Vaswani et al., 2023) treinado com aprendizado por reforço a partir de *feedback* humano (RLHF[2]) (Christiano et al., 2023). O treinamento RLHF consiste de três etapas: primeiro, treinar um modelo de linguagem com aprendizado auto-supervisionado; segundo, reunir dados de comparação baseados em preferências humanas e treinar um modelo de recompensa; e terceiro, otimizar o modelo de linguagem contra o modelo de recompensa através de aprendizado por reforço. Como resultado desse treinamento, o ChatGPT demonstrou

[1] https://platform.openai.com
[2] Reinforcement Learning from Human Feedback.

DOI: 10.21814/lm.16.2.433
This work is Licensed under a Creative Commons Attribution 4.0 License

capacidades impressionantes, como gerar respostas textuais de alta qualidade para entradas humanas, rejeitar perguntas inadequadas e corrigir erros anteriores com base em conversas subsequentes. Após o lançamento do ChatGPT, muitas empresas e pesquisadores iniciaram o desenvolvimento de modelos de linguagem generativos especializados em diferentes idiomas. Para português, por exemplo, o Sabiá é um modelo que foi treinado exclusivamente na língua portuguesa (Pires et al., 2023) e que está disponível através da MariTalk API[3] como um *chatbot*. Embora a maioria dos modelos de linguagem generativos tenha demonstrado capacidades conversacionais impressionantes, a comunidade de PLN ainda está incerta sobre sua capacidade de alcançar uma generalização *zero-shot* ou *few-shot* superior em comparação com os LLMs existentes, especialmente em idiomas diferentes do inglês (Chang et al., 2023). Especificamente, sua eficácia em português do Brasil ainda não foi explorada com maior profundidade.

Para abordar essa lacuna de pesquisa, este trabalho apresenta uma investigação abrangente sobre a capacidade de aprendizado de diferentes modelos LLMs, avaliando os seus desempenhos em uma ampla gama de conjuntos de dados de PLN em português do Brasil, incluindo três tarefas relevantes de análise de sentimentos: i) identificação de sentenças opinativas; ii) cálculo de polaridade; e iii) identificação de sentenças comparativas. Particularmente, trata-se de uma ampliação do trabalho publicado inicialmente no artigo de Araujo et al. (2024). Neste trabalho, ampliou-se a pesquisa com a comparação das técnicas de prompt *zero-shot* e *few-shot*, a introdução da avaliação de uma versão mais recente do ChatGPT (versão 4), a avaliação de um modelo treinado em português, o Sabiá, e a geração de novos resultados de modelos da literatura treinados com dados anotados pela melhor dos LLMs testados.

As três tarefas escolhidas para avaliação são importantes em PLN no que diz respeito a problemas de detecção de informações de comentários de avaliações de pessoas sobre qualquer assunto, de qualquer mídia textual e, principalmente, da Internet. Assim, essas contribuições podem ser aplicadas a vários problemas de mineração de dados. Mais especificamente, foram investidas as seguintes questões de pesquisa neste trabalho:

Questão de Pesquisa 1 (QP1): Como os modelos ChatGPT e Sabiá se comportam como solucionadores para as três tarefas de análise de sentimentos mencionadas acima? Para abordar isso, foi comparado empiricamente o desempenho do ChatGPT e do Sabiá com métodos considerados estado da arte.

Questão de Pesquisa 2 (QP2): Como a anotação gerada pelo melhor modelo de LLM avaliado neste trabalho influencia os dados de treinamento para diferentes classificadores abordando as três tarefas de análise de sentimentos mencionadas acima? Para isso, foi comparada empiricamente a anotação gerada pelo melhor modelo avaliado para dados de treinamento de diferentes classificadores que abordam as três tarefas de análise de sentimentos mencionadas.

Até onde se sabe, este é o primeiro trabalho que investiga o problema de usar um LLM para abordar tarefas relevantes de análise de sentimentos em português. As principais contribuições deste trabalho podem ser resumidas da seguinte forma:

- Realização de experimentos para avaliar o impacto do hiperparâmetro de temperatura no desempenho do ChatGPT em tarefas de PLN.
- Avaliação das versões do ChatGPT 3.5 e ChatGPT 4.0 como solucionadores em três diferentes tarefas de análise de sentimentos. Através da execução dos experimentos, foi possível identificar que o ChatGPT exibe desempenho excepcional em tarefas de análise de sentimentos, especificamente na identificação de subjetividade e polaridade em frases. Em termos de identificação de frases comparativas, o ChatGPT demonstra um desempenho inferior em comparação com os *baselines*.
- Realização de uma análise comparativa entre modelos treinados em múltiplos idiomas, como o ChatGPT, e um modelo treinado exclusivamente com textos em português, como é o caso do Sabiá.
- Análise abrangente da viabilidade de aproveitar o ChatGPT para anotação de dados para tarefas complexas de PLN.

O restante do artigo está organizado da seguinte forma. A Seção 2 apresenta conceitos preliminares importantes para o entendimento do trabalho. A Seção 3 apresenta uma revisão dos trabalhos relacionados sobre Modelos de Linguagem de Grande Escala (LLMs) e a Seção 4 apresenta uma visão geral da metodologia aplicada no presente estudo. A Seção 5 apresente e discute a avaliação experimental da abordagem proposta. Finalmente, a Seção 6 apresenta e discute nossas principais conclusões, limitações e direções futuras de pesquisa.

[3] https://github.com/maritaca-ai/maritalk-api

2. Conceitos Preliminares

2.1. Análise de Sentimentos

Análise de sentimentos é uma subárea do Processamento de Linguagem Natural (PLN) que busca identificar e extrair informações subjetivas de textos, tais como emoções, opiniões e atitudes expressas por usuários em diferentes contextos, incluindo redes sociais, comentários sobre produtos e *feedbacks* de clientes (Liu, 2020).

As tarefas centrais na análise de sentimentos incluem a identificação de características específicas ou atributos dos produtos e serviços mencionados nos textos, conhecida como extração de aspectos (Rana & Cheah, 2016). Estes aspectos extraídos podem ser utilizados em uma miríade de domínios, tais como análise de debate político (Seno et al. (2024)), sumarização de opiniões sobre produtos eletrônicos (de Melo et al., 2018; Hayatin et al., 2024) ou para a identificação de notícias falsas (Hou et al., 2024). Para isso, é necessário realizar a identificação de sentenças opinativas, que se diferenciam de sentenças factuais por expressarem uma opinião ou julgamento (Pandey & Deorankar, 2019; de Oliveira & de Melo, 2021). Essa tarefa de identificação de sentenças opinativas é frequentemente acompanhada pela análise de polaridade, onde a frase é classificada como positiva ou negativa com base na emoção expressa (Oliveira & de Melo, 2020). Por exemplo, em um comentário de um cliente sobre um restaurante, a extração de aspectos permite identificar termos como "comida" ou "atendimento" e associá-los às opiniões positivas ou negativas expressas. Além dessas tarefas, a identificação de sentenças comparativas também é relevante, pois envolve a detecção de frases que comparam dois ou mais elementos, identificando as preferências e percepções dos consumidores (Carvalho et al., 2017). Essas tarefas, em conjunto, fornecem uma visão mais detalhada e específica das opiniões dos usuários, permitindo uma análise mais profunda e direcionada dos sentimentos expressos nos textos.

2.2. Modelos de Aprendizagem de Máquina

A aprendizagem de máquina é um subcampo da Inteligência Artificial que se concentra no desenvolvimento de algoritmos capazes de aprender padrões a partir de dados e realizar previsões ou tomar decisões sem serem explicitamente programados para isso. Esses modelos têm sido amplamente utilizados em uma variedade de aplicações, como reconhecimento de imagem, processamento de linguagem natural e análise preditiva. Entre os modelos mais comuns, o Naive Bayes (NB) se destaca por sua simplicidade e eficácia, especialmente em tarefas de classificação de textos (McCallum & Nigam, 1998). O NB é baseado na aplicação do teorema de Bayes com a suposição de independência entre as características, o que o torna eficiente mesmo em grandes conjuntos de dados. Apesar de sua simplicidade, o NB é robusto e frequentemente utilizado em situações onde a interpretação rápida e a escalabilidade são essenciais.

Outro modelo amplamente utilizado é o Gradient Boosting Trees (GBT), que é um método de aprendizado em conjunto que constrói um modelo preditivo forte a partir de uma combinação de modelos fracos, como árvores de decisão. O GBT funciona iterativamente, ajustando as previsões dos modelos anteriores para minimizar o erro residual, o que o torna particularmente eficaz em tarefas de regressão e classificação (Friedman, 2001). Devido à sua flexibilidade e alto desempenho, o GBT tem sido aplicado com sucesso em áreas como análise de sentimentos, previsão de risco de crédito, detecção de fraudes e outras tarefas que requerem alta precisão preditiva.

Com o aumento da complexidade dos modelos de aprendizado de máquina e a necessidade de otimização de hiperparâmetros, os modelos de *Automated Machine Learning* (AutoML) têm se tornado uma alternativa promissora. O AutoML automatiza o processo de desenvolvimento de modelos, desde o pré-processamento de dados até a seleção do modelo e a otimização de hiperparâmetros, permitindo que os pesquisadores criem modelos robustos e eficientes de maneira mais ágil (Elshawi et al., 2019). Ferramentas como AutoGluon exemplificam essa abordagem, utilizando técnicas avançadas como otimização bayesiana e aprendizado por reforço para explorar o espaço de hiperparâmetros e identificar configurações que maximizam o desempenho do modelo (Erickson et al., 2020). Além disso, AutoGluon integra múltiplos modelos e técnicas de pré-processamento para construir *pipelines* personalizados, otimizados para problemas específicos. Particularmente, para a tarefa de análise de sentimentos em textos, o AutoGluon possui classes como o *TextPredictor* e *TabularPredictor*, que empregam modelos pré-treinados baseados em Transformers, como o ELECTRA, RoBERTa, ou Multilingual BERT, para extração de características de textos, e treinam um modelo agregador para a tarefa de classificação em questão.

Diante do exposto, esses modelos citados têm sido empregados como estado-da-arte em tarefas de análise de sentimentos, e são usados neste artigo como *baseline*.

2.3. Modelos de Linguagem Generativos

Modelos de linguagem generativos são uma classe de modelos de aprendizagem de máquina projetados para gerar texto coerente e relevante em resposta a um determinado *prompt*. Baseados em arquiteturas como as redes neurais do tipo Transformer, esses modelos, como o *Generative Pre-trained Transformer* (GPT), são treinados em grandes quantidades de dados textuais e são capazes de prever a próxima palavra em uma sequência, permitindo-lhes criar sentenças, parágrafos ou até mesmo textos longos que imitam a linguagem humana. Esses modelos têm aplicações em diversas áreas, desde a criação de conteúdo automatizado até a tradução de idiomas, análise de sentimentos e a interação em chatbots.

Um elemento relevante dos modelos de linguagem generativos é o conceito de "temperatura." A temperatura é um parâmetro que controla a aleatoriedade das previsões do modelo. Valores mais baixos (próximos de zero) fazem com que o modelo produza saídas mais determinísticas e conservadoras, enquanto que valores mais altos tornam a saída mais diversificada e criativa. Por exemplo, em uma tarefa onde a precisão é importante, uma temperatura baixa pode ser preferível, enquanto uma tarefa que requer criatividade pode se beneficiar de uma temperatura mais alta.

Além disso, esses modelos podem ser usados em configurações de prompt *zero-shot* e *few-shot*. No modo *zero-shot*, o modelo realiza uma tarefa sem ter sido explicitamente treinado nela, baseando-se apenas no conhecimento adquirido em seu pré-treino e no contexto fornecido pelo prompt. Isso é particularmente útil quando não há dados específicos de treinamento disponíveis para uma tarefa. Já no modo *few-shot*, o modelo recebe alguns exemplos no prompt antes de realizar a tarefa, o que pode ajudar a melhorar a precisão da resposta, permitindo que o modelo ajuste sua compreensão da tarefa com base nesses exemplos. Esses conceitos são fundamentais para explorar o potencial dos modelos de linguagem generativos em uma ampla gama de aplicações.

3. Trabalhos Relacionados

Neste trabalho, investiga-se a capacidade das atuais linguagens generativas em lidar com tarefas clássicas de análise de sentimentos em uma extensa variedade de conjuntos de dados em português do Brasil. Além disso, investiga-se o uso dessas linguagens generativas na tarefa de anotação de dados em substituição ao trabalho humano.

3.1. Tarefas de Análise de Sentimentos

O tema de análise de sentimentos tem despertado o interesse de muitos pesquisadores ao longo dos últimos anos, especialmente para o idioma inglês. Nesse contexto, os trabalhos de Wankhade et al. (2022) e de Tan et al. (2023) fornecem boas referências e um panorama abrangente do estado da arte da área. Apesar de ser menos explorada que para a língua inglesa, a análise de sentimentos em português tem recebido atenção crescente nos últimos anos. Diversos trabalhos têm investigado as principais tarefas envolvidas, como a identificação de subjetividade (de Oliveira & de Melo, 2021), classificação de polaridade (Oliveira & de Melo, 2020) e identificação de sentenças comparativas (Kansaon et al., 2020). Para a tarefa de identificação de polaridade, técnicas que utilizam abordagens baseadas em léxicos (Lazarini et al., 2023; de Melo, 2022), aprendizado de máquina supervisionado (Aires et al., 2018) e, mais recentemente, modelos de linguagem pré-treinados, têm sido amplamente estudadas (Lopes et al., 2021).

Outro aspecto relevante é a tarefa de identificação de subjetividade, que foca em distinguir frases opinativas de factuais. Esta tarefa é essencial para compreender a natureza da opinião expressa em textos. Estudos como o de Oliveira & de Melo (2020) forneceram uma análise pormenorizada sobre a tarefa e conjuntos de dados específicos para o português, auxiliando no desenvolvimento de técnicas mais eficazes para essa tarefa. Adicionalmente, trabalhos sobre a identificação de sentenças comparativas, como o de Kansaon et al. (2020)), exploram a classificação de frases que contêm comparações explícitas ou implícitas entre dois ou mais itens. Esse tipo de análise é particularmente relevante em contextos de comércio eletrônico e avaliações de produtos.

Este trabalho propõe avançar no tema avaliando o impacto de modelos mais recentes, particularmente os modelos de linguagem de larga escala (LLMs), que podem trazer vantagens ao permitir a realização de tarefas de PLN com baixo custo de treino.

3.2. ChatGPT

ChatGPT[4] é um modelo de linguagem generativo desenvolvido pela OpenAI, baseado na arquitetura GPT-3.5, que pode gerar textos coerentes e contextualmente relevantes a partir de um *prompt* (Brown et al., 2020). A OpenAI lançou em setembro de 2022 o ChatGPT 3.5 Turbo e em março de 2023 a versão ChatGPT 4. De acordo com a literatura (Rudolph et al., 2023), o ChatGPT 3.5 Turbo possui 175 bilhões de parâmetros, enquanto a versão ChatGPT 4 possui estimativa de 1,7 trilhões de parâmetros (Schreiner, 2024). Segundo a OpenAI, o ChatGPT pode realizar várias tarefas, como gerar respostas a perguntas, e sumarizar e traduzir textos sem nenhum treinamento adicional. O modelo foi treinado em um grande *corpus* de texto de várias fontes, incluindo livros, artigos e sites.

Desde seu surgimento, vários trabalhos da literatura têm abordado o uso do ChatGPT em tarefas de PLN. De forma geral, explora-se a capacidade do modelo de linguagem de entender e gerar textos por meio de prompts, onde comandos são direcionados a, por exemplo, classificar textos. Esse tipo de pesquisa pode ser observado no trabalho de Zhao et al. (2023), onde tal modelo mostra-se competitivo aos métodos tradicionais para classificar textos relacionados à área de agricultura. De forma similar, Loukas et al. (2023) mostram desempenho similar a modelos não generativos, a custos muito menores, para classificação de textos na área de finanças.

No campo da análise de sentimentos, encontram-se trabalhos mostrando a capacidade desses modelos, como, por exemplo, o de Fatouros et al. (2023), onde a abordagem *zero-shot* do ChatGPT superou até mesmo um modelo especializado na área de finanças, o FinBERT. Já Belal et al. (2023) mostram que, considerando tweets em inglês, o ChatGPT supera técnicas baseadas em análise léxica na tarefa de anotação de dados de sentimentos.

Na tarefa de tradução automática, Com o lançamento do GPT-4, o desempenho do ChatGPT melhorou significativamente, tornando-se comparável a produtos comerciais de tradução, mesmo para idiomas distantes (Jiao et al., 2023).

Várias aplicações de chatbots inteligentes surgiram em diferentes áreas, mostrando, com alguns cuidados, resultados e vantagens significativas (Bahrini et al., 2023). Por exemplo, Sallam et al. (2023) lista as seguintes vantagens da integração do ChatGPT no processo educacional médico: melhoria na aprendizagem personalizada, melhoria no raciocínio clínico e assistência para entender conceitos médicos complexos.

[4] https://openai.com/blog/ChatGPT

3.3. Modelos Generativos em Português

Recentemente, a pesquisa sobre Modelos de Linguagem de Grande Escala (LLMs) tem avançado significativamente. Enquanto muitos dos modelos de linguagem foram treinados em múltiplos idiomas, visando a abrangência e a flexibilidade, estudos recentes têm demonstrado que o pré-treinamento monolíngue pode trazer benefícios substanciais em termos de desempenho para tarefas específicas de um determinado idioma (Pires et al., 2023). Essa abordagem permite que o modelo capture melhor as nuances linguísticas e culturais inerentes à língua alvo. Em particular, para o português do Brasil, a utilização de LLMs otimizados para a língua, como o Sabiá-2, tem mostrado resultados promissores, superando modelos multilíngues em diversas avaliações (Almeida et al., 2024). Particularmente, a arquitetura do Sabiá consiste de modelos Transformers similares ao ChatGPT, no entanto, treinados em conjuntos de dados em português. Este trabalho investiga essa tendência, explorando o potencial e as vantagens de empregar LLMs monolíngues em tarefas de análise de sentimentos no contexto brasileiro, comparando seu desempenho com modelos multilíngues como GPT-3.5 e GPT-4.

Para o objetivo descrito acima, foi escolhido o Sabiá-2 devido às suas capacidades linguísticas e de custo-benefício. O modelo foi desenvolvido especificamente para português e treinado em uma vasta gama de textos no idioma, resultando em uma compreensão mais profunda das nuances linguísticas. Isso é fundamental para tarefas de análise de sentimentos, onde a interpretação correta de emoções e sentimentos é crucial. Os resultados do Sabiá-2, conforme descrito por Almeida et al. (2024), mostraram que ele supera ou iguala modelos avançados como GPT-3.5 e GPT-4 em diversos exames, destacando sua eficácia. O modelo Sabiá-2 está disponível, seguindo padrão disponibilizado pela OpenAI, através de uma API nomeada de MariTalk.

A escolha do Sabiá-2 também se justifica pela sua vantagem econômica. O Sabiá-2 Medium, por exemplo, oferece um desempenho comparável ao GPT-4 a um custo significativamente menor por *token*. Esta característica torna o Sabiá-2 uma opção viável para aplicações de análise de sentimentos em larga escala, onde a relação custo-eficácia é um fator crucial. Além disso, o foco do Sabiá-2 em um corpus monolíngue e es-

pecializado no português permite uma integração mais eficiente de conhecimentos específicos do domínio, o que é essencial para capturar corretamente as nuances culturais e contextuais presentes nas expressões sentimentais do português brasileiro. Em suma, o Sabiá-2 não só pode proporcionar uma análise de sentimentos mais precisa, mas também o faz de maneira mais econômica, justificando assim sua escolha para este estudo.

3.4. Anotadores

Em aplicações de PLN, o uso de dados rotulados é frequentemente necessário, o que envolve o processo manual de anotação de dados. Tradicionalmente, duas estratégias principais têm sido empregadas para esse fim. Primeiramente, pesquisadores podem recrutar e treinar codificadores, como assistentes de pesquisa, para realizar a tarefa de anotação. Uma outra forma de anotação consiste em confiar nas pessoas que se disponibilizam a realizar essa tarefa em plataformas como Amazon Mechanical Turk (MTurk) para anotar os dados (Gilardi et al., 2023).

Em uma análise recente (Gilardi et al., 2023), foi demonstrado que o ChatGPT superou trabalhadores humanos na anotação de texto em várias tarefas. Além disso, outros estudos (Ding et al., 2022) mostraram que o desempenho dos modelos ChatGPT é ligeiramente inferior quando comparado a dados rotulados por humanos. No entanto, a utilização de modelos ChatGPT reduz significativamente o custo e o tempo necessários para o processo de anotação em comparação com a dependência exclusiva de anotadores humanos.

Particularmente, trabalhos como os apresentados por Qin et al. (2023) têm objetivos semelhantes aos objetivos desta pesquisa; no entanto, eles estão principalmente focados na língua inglesa. Em contraste, este trabalho fornece uma contribuição adicional ao avaliar o desempenho de modelos ChatGPT em textos em português.

Essas descobertas indicam que o ChatGPT apresenta capacidades promissoras em realizar a tarefa de anotação de dados de texto com muitas vantagens, como desempenho ou custos, quando comparado à dependência exclusiva de anotadores humanos. Por essas razões, decidiu-se investigar o uso do ChatGPT na geração automática de dados de treinamento (QP2).

4. Metodologia

O objetivo principal deste estudo é investigar o potencial de generalização dos modelos de linguagem de grande escala (LLMs) em diversas tarefas de análise de sentimentos, especificamente no contexto do português brasileiro. Esta pesquisa se concentra em duas questões principais.

A primeira questão de pesquisa (QP1) busca validar empiricamente o desempenho dos modelos de linguagem escolhidos como solucionadores competentes para tarefas relevantes de análise de sentimentos. Para validar esta questão de pesquisa, foram conduzidas avaliações em três tarefas cruciais de análise de sentimentos descritas a seguir. Além de adotar a abordagem *zero-shot* de usar os modelos de linguagem para classificação dos textos, foi avaliado o impacto da abordagem *few-shot* de engenharia de prompt.

A primeira tarefa (Tarefa 1) consiste em classificar sentenças como factuais ou opinativas. O design do prompt para essa tarefa é mostrado na Figura 1 (a). Por exemplo, a frase "o restaurante tem um ambiente agradável" seria classificada como opinativa, enquanto a sentença "o restaurante abre às 14 horas" seria classificada como factual. Este estudo adotou como base a metodologia descrita por de Oliveira & de Melo (2021) e também utilizou os conjuntos de dados disponibilizados pelos autores desse artigo. No trabalho de de Oliveira & de Melo (2021), os autores analisaram diversos modelos clássicos de aprendizado de máquina, e o modelo GBT alcançou o melhor resultado ao utilizar a classificação de tags de partes do discurso (POS tags) das palavras, sendo o número de adjetivos a principal característica.

A segunda tarefa (Tarefa 2) tem como objetivo principal classificar cada sentença como possuindo sentimento positivo ou negativo. O *design* do prompt para esta tarefa é mostrado na Figura 1 (b). A frase "a comida estava deliciosa" expressa um sentimento positivo, enquanto "o preço era muito salgado" transmite um sentimento negativo sobre o preço do restaurante. As metodologias elaboradas por Oliveira & de Melo (2020) foram empregadas como base para esta tarefa, e os conjuntos de dados publicados pelos respectivos autores também foram utilizados. No trabalho de Oliveira & de Melo (2020), os autores compararam diversos métodos que representavam o estado da arte no tratamento de cálculo de polaridade de texto em português com um classificador GBT, utilizando as seguintes características: número de palavras, adjetivos, substantivos, advérbios, superlativos, comparativos e termos modificadores de substantivos. O classificador GBT apresentou resultado superior e será adotado como modelo de referência (*baseline*) nesta tarefa. Neste trabalho, assim como no trabalho de Oliveira & de Melo (2020)) foram consi-

deradas apenas os sentimentos positivos e negativos. Apesar de alguns trabalhos na literatura considerarem a classificação de sentenças neutras, estas costumam ter uma menor relevância (Wankhade et al., 2022), pois o maior interesse é na descoberta de quais sentenças transmitem um sentimento positivo ou negativo.

A terceira tarefa (Tarefa 3) consiste em classificar sentenças como comparativas ou diretas. O *design* do prompt para esta tarefa é mostrado na Figura 1 (c). Por exemplo, a frase "o restaurante tem um ambiente agradável" é uma frase direta, enquanto a frase "o sorvete da McDonald's é melhor" é comparativa. Os métodos descritos por Kansaon et al. (2020) serviram como base para esta tarefa, e os conjuntos de dados publicados pelos autores também foram utilizados. Neste trabalho, os autores analisaram diversos modelos clássicos de aprendizado de máquina aplicados em combinação com três diferentes características textuais: Frequência do Termo e da Frequência Inversa do Documento (TF-IDF[5]) de palavras, TF-IDF de bigramas de palavras e TF-IDF de trigramas de palavras, sendo que o modelo Naive Bayes (NB) alcançou o melhor resultado.

A segunda questão de pesquisa (QP2) tem como objetivo validar a viabilidade do uso de modelos ChatGPT para automatizar a rotulagem de conjuntos de dados. Inicialmente, foi empregado o ChatGPT para rotular os dados obtidos na QP1. Em seguida, os dados rotulados pelo ChatGPT foram utilizados para treinar modelos usando o AutoGluon (Erickson et al., 2020). Por fim, os resultados obtidos desses modelos foram comparados, tanto com *baselines* da literatura quanto com o próprio ChatGPT, para avaliar seus desempenhos e eficácia.

4.1. Exploração de Modelos de Linguagem

A OpenAI disponibiliza uma ampla gama de modelos através de sua API, cada um voltado para propósitos e *benchmarks* de performance distintos. Neste estudo, foram avaliados o GPT 3.5-Turbo, o Modelo de Linguagem Grande (LLM) com 175 bilhões de parâmetros, que também é a base do ChatGPT *online* - referenciado como ChatGPT 3.5 e em sua versão mais recente e comercial, o ChatGPT 4.0. Estes modelos se destacam pelo pioreirismo e revolução causada na área de Processamento de Linguagem Natural, e são otimizados para funcionalidades de chat, tornando-os ideal para tarefas centradas na interação por diálogo. Os experimentos foram realizados através da API oficial da OpenAI, com os mesmos parâmetros e versão do modelo, a menos que especificados de outra forma.

Para avaliar o impacto do parâmetro de temperatura dos modelos, que controla o grau de aleatoriedade da saída do modelo, as tarefas foram executadas com o valor de 0, que implica maior determinismo, e também com o valor de 1.0, que implica maior aleatoriedade. Conforme observado por Gilardi et al. (2023), o uso de valores de temperatura mais baixos resulta em melhores resultados do ChatGPT na tarefa de análise de sentimentos.

4.2. Prompts

De acordo com Liu et al. (2023), um prompt funciona como um conjunto de instruções fornecidas a um modelo de linguagem, efetivamente programando-o por meio da customização, aprimoramento ou refinamento de suas capacidades. Selecionar um prompt adequado é essencial para que o modelo forneça a resposta desejada com precisão. Inicialmente, foram experimentados prompts com instruções mais detalhadas, mas se observou que prompts com instruções diretas apresentaram melhores resultados. Abaixo, apresenta-se o prompt selecionado para cada tarefa seguindo a abordagem *zero-shot* descrita na Figura 1.

Para a Tarefa 1, foi escolhido o seguinte prompt: *Classifique a sentença* "`FRASE`" *em factual ou opinativa. Responda somente factual ou opinativa*, onde a frase que se quer avaliar fica entre aspas simples. Com este prompt, espera-se que o modelo responda apenas com as palavras "*factual*" ou "*opinativa*."

Para a Tarefa 2, foi escolhido o seguinte prompt: *Classifique a sentença* "`FRASE`" *em positiva ou negativa. Responda somente positiva ou negativa*, onde a frase que se quer avaliar fica entre aspas simples. Com este prompt, espera-se que o modelo responda apenas com a palavra "*positiva*" ou "*negativa*."

Por fim, para a Tarefa 3, foi escolhido o seguinte prompt: *Classifique a sentença* "`FRASE`" *em comparativa ou não comparativa. Responda somente comparativa ou não comparativa*, onde a frase que se quer avaliar fica entre aspas simples. Com este prompt, espera-se que o modelo responda apenas com a palavra "*comparativa*" ou "*não comparativa*."

[5]Em inglês, TF-IDF significa *Term Frequency-Inverse Document Frequency*.

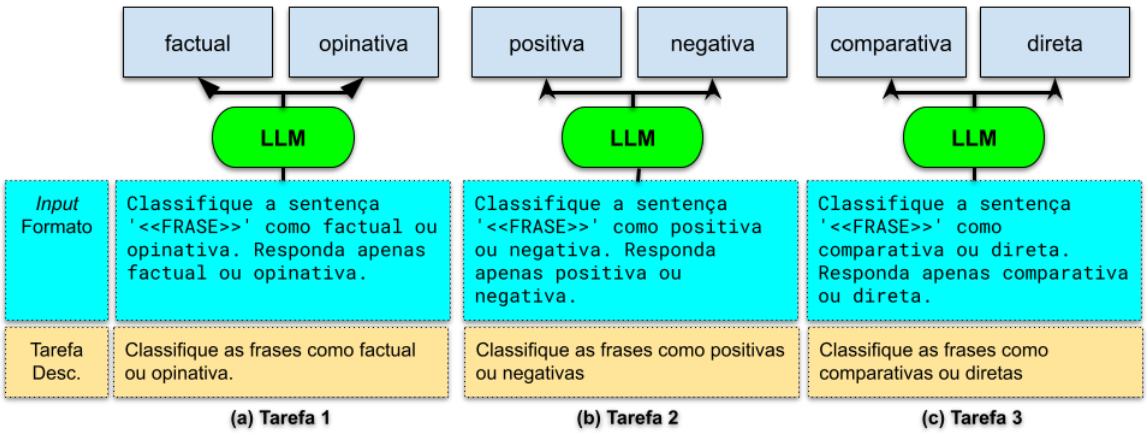


Figura 1: Design de prompts *zero-shot* (ZS).

Para a avaliação dos modelos seguindo a abordagem *few-shot*, adotou-se o *design* de prompt da Figura 2. Nela, as chaves "`classe_1`" ou "`classe_2`" são substituídas conforme as tarefas descritas anteriormente. A chave "`sentença`" é a frase sendo avaliada. E as chaves "`exemplo_classe_1`" e "`exemplo_classe_2`" são frases escolhidas aleatoriamente dentre os datasets avaliados, sendo uma de cada classe para a respectiva tarefa. As frases de exemplo são apresentados nas tabelas 1, 2 e 3, conforme *datasets* usados e melhor descritos na próxima seção.

5. Experimentos

Nesta seção, detalha-se o ambiente de avaliação, o que inclui a descrição dos dados utilizados (*dataset*) e as métricas de avaliação adotadas. Em seguida, os resultados experimentais são apresentados e discutidos.

5.1. Datasets

Diversos conjuntos de dados foram empregados para testar a robustez das linguagens generativas em face de diversos desafios linguísticos e contextuais inerentes ao português brasileiro, garantindo uma validação abrangente para tarefas variadas de análise de sentimentos e alinhamento com *benchmarks*.

Para a Tarefa 1, foram utilizados três conjuntos de dados distintos compostos por sentenças factuais e subjetivas. Os detalhes de cada conjunto de dados empregados na Tarefa 1 são apresentados na Tabela 4. ReLi[6] consiste em uma coleção de resenhas de livros em português, recuperadas da internet e anotadas manualmente (Freitas et al., 2012). Neste trabalho, foi utilizado uma quantidade menor de instâncias do ReLi, pois representa o conjunto de dados empregados nos experimentos do artigo adotado como referência (Oliveira & de Melo, 2020). TA-Restaurantes[7] contém sentenças em português relacionadas a comentários de restaurantes coletados do TripAdvisor[8] (Oliveira & de Melo, 2020). Computer-BR[9] é um conjunto de tuítes em português e abrange uma ampla gama de tópicos relacionados a computadores (Moraes et al., 2016).

Para a Tarefa 2, foram utilizados os mesmos conjuntos de dados da Tarefa 1. No entanto, anotações adicionais para polaridade de sentimento (positiva ou negativa) foram adicionadas. Além disso, foi considerando também o *corpus* do Google Play[10] (Stiilpen Junior & Merschmann, 2016)). Este *corpus* consiste em 1.630 frases selecionadas aleatoriamente de um conjunto original de 10.000 avaliações de aplicativos móveis na Google Play Store. As frases do *corpus* do Google Play são divididas igualmente entre sentimentos positivos e negativos. Os detalhes de cada conjunto de dados empregados na Tarefa 2 são apresentados na Tabela 5.

Para a Tarefa 3, foram utilizados dois conjunto de dados sumarizados na Tabela 6. O Twitter é um *corpus* de frases comparativas relacionadas a produtos eletrônicos (Kansaon et al., 2020) e o Buscapé consiste em avaliações de produtos coletadas do site Buscapé[11] (Kansaon et al., 2020). Os conjuntos de dados são anotados como sentenças comparativas ou diretas.[12]

[6] https://www.linguateca.pt/Repositorio/ReLi

[7] https://data.mendeley.com/datasets/hsn6g3dbsk

[8] https://www.tripadvisor.com.br

[9] http://tiagodemelo.info/datasets.html

[10] http://tiagodemelo.info/datasets.html

[11] https://www.buscape.com.br

[12] https://zenodo.org/records/4124410

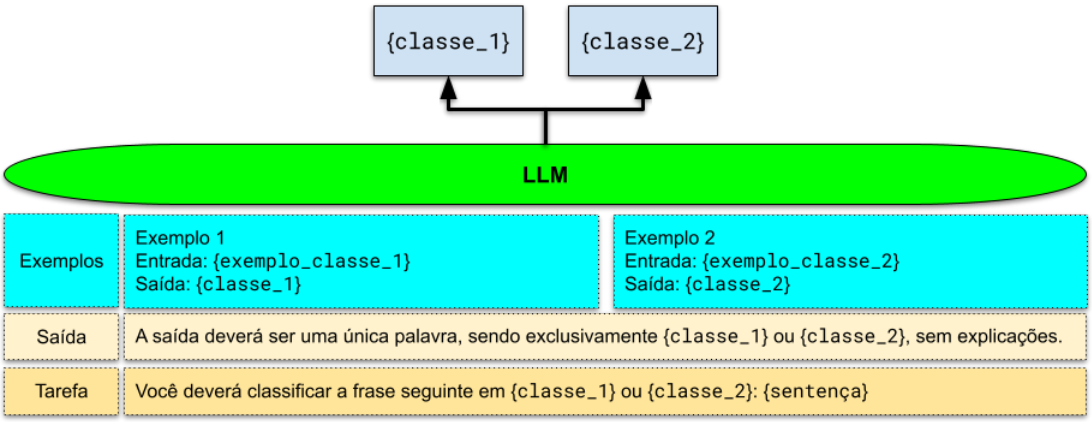


Figura 2: Design de prompts *few-shot* (FS).

Dataset	Exemplos
ReLi	factual: "Indicado para crianças e pré-adolescentes!" opinativa: "Não tenho palavras para explicar o quanto eu amei a experiência de ler finale e todo hush hush. Vou guardar essa história comigo para sempre."
TA-Restaurants	factual: "Vários tipos de molho regional, com tucupi e outros." opinativa: "Os pratos, petiscos e bebidas são nota 10!"
Computer-Br	factual: "Notebook Dell Vostro V14T-5470-B30 Intel Core i5 4GB (2GB de Memória Dedicada) 500GB LED 14"Touchscreen Windows 8... http://fb.me/J7liX8dJ" opinativa: "Amooo meu notebook #Dell #corei7 ????"

Tabela 1: Exemplos usados em prompts *few-shot* para a Tarefa 1.

Dataset	Exemplos
ReLi	positiva: "Eu adorei esse livro quando li, foi um conselho de um garoto que é escritor eu conheci em a internet, foi ótimo." negativa: "Acho que este livro não merece nem mesmo uma estrela."
TA-Restaurants	positiva: "Ambiente amplo, confortável e muito agradável." negativa: "atendimento deixa a desejar."
Computer-Br	positiva:" Selok meo not é mol rapido kk DELL é zikaa" negativa: "Reclame Aqui - Dell Computadores do Brasil - Dell erra na entrega e não cumpre prazo http://fb.me/6oW8KuPSt"
Google Play	positiva: "Execelente Muito bom esse jogo!!! Adoreei...recomendo." negativa: "Max O grafico é ruim"

Tabela 2: Exemplos usados em prompts *few-shot* para a Tarefa 2.

Dataset	Exemplos
Buscapé	não-comparativa: "completa tem suas vantagens mas fui lezado comprei esta tv no dia 26"' comparativa: "tenho uma brastemp 11 kg bwa11 que é muito inferior a essa consul."'
Twitter	não-comparativa: "já tentei assistir black mirror três vezes mas paro no 2 ep pq fico sem entender nada!"' comparativa: "a glock ou vc escolho a glock"'

Tabela 3: Exemplos usados em prompts *few-shot* para a Tarefa 3.

	Factual	Subjectiva	Total
ReLi	175	175	350
TA-Restaurantes	591	458	1.049
Computer-BR	604	1.677	2.281

Tabela 4: *Dataset* para a Tarefa 1.

	Positivo	Negativo	Total
ReLi	85	85	170
TA-Restaurantes	505	56	561
Computer-BR	198	400	598
Google Play	815	815	1.630

Tabela 5: *Dataset* para a Tarefa 2.

	Direta	Comparativa	Total
Buscapé	1.282	1.472	2.754
Twitter	918	1.135	2.053

Tabela 6: *Dataset* para a Tarefa 3.

5.2. Métricas de Avaliação

Para avaliar os modelos nas tarefas investigadas neste artigo, foram utilizadas as métricas de precisão (P), revocação (R) e F-measure (F1) (Baeza-Yates & Ribeiro-Neto, 1999). Seja A o conjunto de respostas corretas, de acordo com um conjunto de referência, e seja B o conjunto de respostas produzidas pelo método que está sendo avaliado. Definiu-se precisão (P), revocação (R) e F-score (F_1) como:

$$P = \frac{|A \cap B|}{|B|} \quad R = \frac{|A \cap B|}{|A|} \quad F_1 = \frac{2 \times (P \times R)}{P + R}$$

5.3. Resultados

Nesta seção, são apresentados os resultados das questões de pesquisa levantadas QP1 e QP2 para os diferentes *datasets* e modelos das três tarefas.

5.3.1. Questão de Pesquisa 1 (QP1)

Inicialmente, foi avaliada a influência do hiperparâmetro temperatura no desempenho em todas as tarefas. Para isso, foi escolhido o ChatGPT 3.5 como referência, por ser o modelo de acesso livre e amplamente utilizado em pesquisas com LLMs. Foram consideradas a temperatura de 0, onde o modelo é totalmente determinístico, e a temperatura de 1, onde o modelo gera respostas mais criativas. A Figura 3 exibe os valores do score F1 para as diferentes tarefas (em cores diferentes) e para cada conjunto de dados de uma determinada tarefa. É importante notar que o modelo com temperatura 0 produziu resultados melhores ou, no mínimo, iguais ao modelo com temperatura 1. A justificativa para isso é que o objetivo da classificação de texto é produzir uma saída única para uma dada entrada. Portanto, a liberdade de escolher respostas mais variadas e criativas tende a gerar resultados piores em tarefas de classificação de texto.

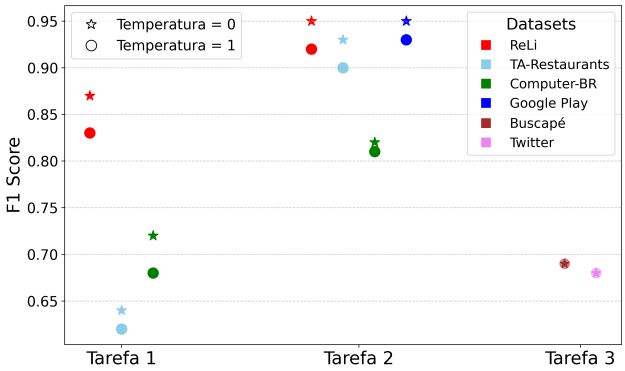


Figura 3: Desempenho do ChatGPT 3.5 conforme hiperparâmetro de temperatura.

Os resultados para todas as tarefas, descritos a seguir, têm os modelos com temperatura igual a zero, e são comparados com métodos da literatura em cada uma das tarefas avaliadas. Os modelos configurados na abordagem *zero-shot* estão representados por ZS e os modelos configurados na abordagem *few-shot* estão representados por FS.

Os resultados para a Tarefa 1 (identificação de subjetividade) são apresentados na Tabela 7. A análise mostra que, na configuracão de prompt *zero-shot* (indicado por ZS nos modelos) e no *dataset* ReLi, os modelos de linguagem ChatGPT 4, ChatGPT 3.5 e MariTalk alcançaram, nesta ordem, resultados muito próximos ao GBT, que é o estado da arte para esta tarefa. Todos os modelos de linguagem apresentaram desempenho inferior no *dataset* TA-Restaurantes, com destaque negativo para o MariTalk. Já no *dataset* Computer-BR, a situação se inverteu, onde todos os modelos de linguagem superaram o GBT, com destaque positivo para o MariTalk. Considerando a metodologia de prompt *few-shot* (indicado por FS nos modelos), pode-se observar que não houve ganhos nos modelos ChatGPT 3.5 e ChatGPT 4 em relação às suas versões *zero-shot*, mas o desempenho do MariTalk subiu consideravelmente, superando o GBT nos *datasets* ReLi e Computer-BR.

	ReLi			**TA-Restaurantes**			**Computer-BR**		
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1
GBT	0,76	0,68	0,71	**0,71**	**0,91**	**0,80**	0,39	0,34	0,36
ChatGPT 3.5$_{ZS}$	0,68	0,68	0,68	0,63	0,63	0,63	0,54	0,54	0,54
ChatGPT 4$_{ZS}$	0,69	0,69	0,69	0,64	0,64	0,64	0,58	0,58	0,58
MariTalk$_{ZS}$	**0,82**	0,57	0,67	0,47	0,39	0,43	0,78	0,73	0,75
ChatGPT 3,5$_{FS}$	0,58	0,58	0,58	0,59	0,59	0,59	0,32	0,32	0,32
ChatGPT 4$_{FS}$	0,63	0,63	0,63	0,62	0,62	0,62	0,54	0,54	0,54
MariTalk$_{FS}$	**0,82**	**0,79**	**0,81**	0,47	0,46	0,46	**0,79**	**0,76**	**0,78**

Tabela 7: Identificação de subjetividade (Tarefa 1).

Em relação à abordagem de prompt *few-shot* (FS), percebe-se que houve uma piora nos modelos ChatGPT 3.5 e 4 em relação às versões *zero-shot* (ZS). Isso pode ser explicado pelo aumento do tamanho dos prompts, causando mais dificuldades de interpretação dos mesmos sem agregar novos conhecimentos. No entanto, o modelo MariTalk se beneficiou pelo prompt com exemplos da tarefa. De fato, observou-se que o MariTalk na versão ZS produziu muitos erros de formatação em sua saída (palavras ou formatações diferentes das anotações de saída solicitadas, causando erros de classificação). Isso se observa pela diferença entre as métricas de precisão e revocação em todos os *datasets*, indicando a ocorrência de muitos falsos negativos. Neste modelo, a abordagem *few-shot* ajudou a produzir saídas melhor formatadas, melhorando o resultado geral e aproximando as métricas de precisão e revocação do modelo.

Em relação às diferenças de desempenho nos diferentes *datasets*, nota-se que tanto o ReLi quanto o TA-Restaurantes são compostos por textos mais descritivos e formais se comparados ao Computer-BR, que é composto por tuítes. Esses tuítes são, geralmente, escritos de forma abreviada, utilizando jargões ou linguagens coloquiais. Assim, pode-se ver que o modelo da literatura, treinado especificamente para o *dataset*, teve um desempenho muito melhor nos dois primeiros casos. No entanto, os modelos de linguagem mostraram-se mais resilientes aos dados ruidosos do último conjunto. Ao se analisar os resultados dos experimentos, percebe-se que as versões do ChatGPT podem não entender bem a subjetividade de uma frase na maioria dos casos, mas é muito mais capaz de lidar com diferentes tipos de texto devido à grande quantidade de dados e a diversidade de textos usados em seu treinamento. Já o MariTalk, parece ter se beneficiado de sua base de treinamento de textos em português, mostrando superioridade evidente no ReLi e no Computer-BR.

Os resultados para a Tarefa 2 (identificação de polaridade) são apresentados na Tabela 8. Os dados tabulados mostram que os modelos de linguagem alcançaram resultados significativamente melhores que o modelo GBT, que representa o estado da arte, nos conjuntos de dados ReLi, Computer-BR e Google Play, enquanto apresentaram um F1-score similar no TA-Restaurants. O ChatGPT 4 foi melhor que os demais, obtendo os maiores valores de F1-score em sua versão *zero-shot*. Já o MariTalk perdeu para as versões do ChatGPT na maioria dos *datasets*, ficando com desempenho bem próximo no TA-Restaurants. A abordagem de prompts *few-shot* apresentou pouca vantagem na maioria dos casos, tendo ajudado levemente o desempenho do ChatGPT 3.5 nos *datasets* Computer-BR e Google Play, mas ao contrário da Tarefa 1, prejudicando mais o MariTalk, como observado no F1 desses mesmos dados.

Os resultados sugerem que os modelos de linguagem e, em especial, o ChatGPT, são altamente capazes de determinar a polaridade das sentenças. Mesmo não sendo ajustados para esses conjuntos de dados específicos, é plausível que a análise de sentimentos e polaridade seja comum nos diversos textos utilizados para o treinamento desses modelos. Por exemplo, é esperado que textos de conversas e literatura abordem a positividade ou não das ideias muito mais do que a subjetividade. Além disso, o treinamento dos ChatGPTs incorporou avaliações de usuários relacionadas a produtos e serviços de diversas plataformas. Esse *feedback* geralmente inclui um sistema de classificação por estrelas: comentários com 1 ou 2 estrelas são interpretados como negativos, enquanto comnentários com 4 ou 5 estrelas são positivos. Isso permite aos modelos discernir efetivamente a polaridade de termos e frases dentro dessas avaliações. Essas observações podem ajudar a esclarecer a dificuldade maior dos modelos de linguagem na Tarefa 1. Por último, prompts que buscam o sen-

	Reli			**TA-Restaurants**			**Computer-BR**			**Google Play**		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
GBT	0,47	0,64	0,59	0,90	**0,99**	**0,95**	0,44	0,44	0,44	0,69	0,68	0,69
ChatGPT 3.5$_{ZS}$	**0,96**	**0,96**	**0,96**	0,93	0,93	0,93	0,82	0,82	0,82	0,95	0,95	0,95
ChatGPT 4$_{ZS}$	**0,96**	**0,96**	**0,96**	**0,94**	0,94	0,94	**0,92**	**0,92**	**0,92**	**0,98**	**0,98**	**0,98**
MariTalk$_{ZS}$	0,85	0,85	0,85	**0,94**	0,94	0,94	0,89	0,89	0,89	0,94	0,94	0,94
ChatGPT 3.5$_{FS}$	0,95	0,95	0,95	0,93	0,93	0,93	0,89	0,89	0,89	0,97	0,97	0,97
ChatGPT 4$_{FS}$	**0,96**	**0,96**	**0,96**	**0,94**	0,94	0,94	0,91	0,91	0,91	0,97	0,97	0,97
MariTalk$_{FS}$	0,82	0,82	0,82	0,93	0,93	0,93	0,69	0,69	0,69	0,82	0,82	0,82

Tabela 8: Identificação de polaridade (Tarefa 2).

timento do texto tendem a ser mais diretos do que aqueles que investigam a subjetividade (factual ou opinativo). Essa clareza intrínseca nos prompts de sentimento pode reduzir as chances de interpretações equivocadas pelo modelo.

Os resultados para a Tarefa 3 (identificação de frases comparativas) são apresentados na Tabela 9. Todos os modelos de linguagem apresentaram desempenho inferior ao método NB, que é o estado-da-arte. Mas o ChatGPT 4 apresentou um desempenho superior em relação aos demais modelos de linguagem, mostrando uma evolução de entendimento em relação à sua versão anterior. Já o MariTalk, mesmo treinado em português, teve desempenho similar ao ChatGPT 3.5. Novamente, a abordagem *few-shot* não se mostrou significativa, tendo ajudado levemente o ChatGPT 4 no *dataset* Twitter, mas deteriorando o desempenho dos modelos nos demais casos.

	Buscape			**Twitter**		
	P	R	F_1	P	R	F_1
NB	**0,87**	**0,88**	**0,87**	**0,86**	**0,86**	**0,86**
ChatGPT 3.5$_{ZS}$	0,67	0,67	0,67	0,61	0,61	0,61
ChatGPT 4$_{ZS}$	0,75	0,75	0,75	0,70	0,70	0,70
MariTalk$_{ZS}$	0,67	0,67	0,67	0,68	0,68	0,68
ChatGPT 3.5$_{FS}$	0,51	0,51	0,51	0,58	0,58	0,58
ChatGPT 4$_{FS}$	0,75	0,75	0,75	0,73	0,73	0,73
MariTalk$_{FS}$	0,47	0,46	0,46	0,48	0,47	0,48

Tabela 9: Identificação de sentenças comparativas (Tarefa 3).

Diferentemente das Tarefas 1 e 2, onde modelos de linguagem generativos tiveram desempenhos competitivos ou superiores, foram observadas maiores diculdades desses no reconhecimento de frases comparativas. Essa limitação possivelmente se deve ao fato desses modelos não terem sido treinado nesses conjuntos de dados específicos. Além disso, é possível que os textos comuns utilizados durante seu treinamento não apresentem frequentemente julgamentos comparativos explícitos, ponto já discutido no contexto da Tarefa 1, e que contrasta com as expectativas para a Tarefa 2. Por exemplo, frases como "acho um ótimo smartphone em relação ao seu preço com muitas funções" e "preço poderia ser mais acessível já que a Caloi é no brasil" são identificadas como frases comparativas pelo ChatGPT 3.5, apesar de não haver comparação explícita entre dois produtos.

Como pode-se observar, os modelos de linguagem demonstraram elevado desempenho na análise de sentimentos, particularmente na identificação de subjetividade e polaridade dentro das frases. Na tarefa de identificação de polaridade, o desempenho do ChatGPT 4 se destaca como o melhor no geral, sugerindo que ele pode lidar com essas tarefas de forma confiável. Para a identificação de frases comparativas, embora os modelos de linguagem não tenham obtido os melhores resultados, é possível observar uma evolução do ChatGPT em sua versão 4. Ainda, a seleção de um prompt mais adequado poderia melhorar os resultados. A adição de mais *tokens* poderia refinar ainda mais as respostas, mas isso traria a contra-partida de aumentar o custo por requisição. O MariTalk mostrou-se competitivo com o ChatGPT 3.5, ultrapassando seu desempenho em vários casos, o que evidencia o potencial de um modelo treinado exclusivamente em português para tarefas nessa língua. Os resultados experimentais indicam que o ChatGPT 4 poderia ser utilizado como um método adequado para abordar as tarefas analisadas.

Na Tabela 10, pode-se observar ainda, alguns exemplos de frases que os modelos não conseguiram classificar corretamente, algumas frases apresentam ambiguidade, o que dificultaria o desempenho do modelo. Outras frases como, por exemplo, "meu pai é fofin dms, diferente da minha mãe" demonstram que apesar de serem bons classificadores, os LLMs ainda não são capazes de entender perfeitamente as nuances do idioma português.

Datasets	Exemplos
ReLi	"Em o final, me envolvi, me vi vendo tudo que os personagens não viam" Resposta (ChatGPT 3.5): negativa Resposta esperada: positiva
TA-Restaurantes	"Sucos com frutas locais além dos tradicionais." Resposta (ChatGPT 3.5): positiva Resposta esperada: negativa
Computer-BR	"Entro no site da dell com a intenção de comprar o notebook mais barato, aí eles fazem a comparação com os mais caros..." Resposta (ChatGPT 3.5): positiva Resposta esperada: negativa
Google Play	"Bacana Bom na hora do almoço :(" Resposta (ChatGPT 4): negativa Resposta esperada: positiva
Buscapé	"outros smartphones android de parecido não possuem" Resposta (ChatGPT 3.5): comparativa Resposta esperada: não-comparativa
Twitter	"meu pai é fofin dms, diferente da minha mãe" Resposta (ChatGPT 3.5): não-comparativa Resposta esperada: comparativa

Tabela 10: Exemplos de anotações incorretas dos modelos.

Foi ainda realizada uma análise comparativa do número de *tokens* e do custo da realização dos experimentos com as linguagens generativas. A Tabela 11 apresenta a quantidade total de tokens usados nas tarefas, considerando as abordagens *zero-shot* e *few-shot*. Os valores apresentados dentro dos parênteses representam a média do número de *tokens* por sentença em cada domínio.

Tarefas	Datasets	zero-shot	few-shot
Tarefa 1	ReLi	20.574 (59)	59.043 (169)
	TA-Restaurantes	47.740 (46)	143.084 (136)
	Computer-BR	145.312 (64)	436.985 (192)
Tarefa 2	ReLi	10.205 (60)	28.552 (168)
	TA-Restaurantes	26.765 (48)	72.716 (130)
	Computer-BR	35.427 (59)	104.716 (175)
	Google Play	79.286 (49)	219.067 (134)
Tarefa 3	Buscapé	139.182 (51)	421.895 (153)
	Twitter	102.807 (50)	301.346 (147)

Tabela 11: Número de *tokens* empregados nos experimentos.

É possível observar que a abordagem *few-shot* usou cerca de três vezes mais *tokens* quando comparada com a abordagem *zero-shot*. Essa quantidade a mais resulta em um maior tempo de processamento e também em um custo mais elevado. A Tabela 12 apresenta uma estimativa de custos[13] no uso das linguagens generativas tomando-se como referências os custos apresentados nos sites do ChatGPT e MariTalk. Observa-se que apesar do modelo ChatGPT ter obtido resultados superiores na maioria das tarefas, o seu custo é mais que o triplo da sua versão 3.5 e o dobro do MariTalk. Com base nesses valores, é possível concluir que a depeneder da tarefa e do volume de dados que precisem ser processados, usar modelos anteriores ou um modelo nacional poderiam trazer resultados próximos ao estado da arte, mas com um custo bastante reduzido.

	zero-shot	few-shot
ChatGPT 3.5	1,2 dólares	3,4 dólares
ChatGPT 4	3,6 dólares	10,2 dólares
MariTalk	1,8 dólares	5,1 dólares

Tabela 12: Estimativa dos custos no uso dos LLMs.

5.3.2. Questão de Pesquisa 2

O objetivo da QP2 é verificar experimentalmente se a classificação de sentenças por modelos de linguagem pode ser usada para treinar um modelo AutoML. Os resultados apresentam a comparação entre os modelos da literatura, com o ChatGPT 4_{ZS}, na configuração de prompts *zero-shot*, que obteve os melhores resultados na avaliação anterior, e o AutoGluon. Para isso, utilizou-se o ChatGPT 4_{ZS} para anotar todos

[13]Estes custos foram calculados a partir de valores publicados no dia 06 de junho de 2024.

	Reli			**TA-Restaurantes**			**Computer-BR**		
	P	R	F_1	P	R	F_1	P	R	F_1
GBT	0.76	0.68	0.71	**0.71**	**0.91**	**0.80**	0.39	0.34	0.36
ChatGPT 4_{ZS}	0.69	0.69	0.69	0.64	0.64	0.64	0.58	0.58	0.58
AutoGluon	**0.88**	**0.86**	**0.86**	0.68	0.61	0.60	**0.73**	**0.79**	**0.72**

Tabela 13: Identificação de subjetividade (Tarefa 1) - usando ChatGPT 4_{ZS} como anotador.

	ReLi			**TA-Restaurantes**			**Computer-BR**			**Google Play**		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
GBT	0.57	0.64	0.59	0.90	**0.99**	**0.95**	0.44	0.44	0.44	0.69	0.68	0.69
ChatGPT 4_{ZS}	**0.96**	**0.96**	**0.96**	**0.94**	0.94	0.94	**0.92**	**0,92**	**0.92**	**0.98**	**0.98**	**0.98**
AutoGluon	0.90	0.86	0.87	0.76	0.64	0.68	0.91	0.81	0.84	0.93	0.92	0.92

Tabela 14: Identificação de polaridade (Tarefa 2) - usando ChatGPT$_{ZS}$ como anotador.

os datasets das três tarefas consideradas neste trabalho (seção 5.1). O AutoGluon foi treinado com esses rótulos e avaliou-se as suas métricas de treino de forma comparativa aos resultados dos demais modelos perante as anotações originais.

A Tabela 13 mostra os resultados comparativos para a identificação de subjetividade (Tarefa 1). Observa-se que o desempenho do AutoGluon nos *datasets* ReLi e Computer-BR superou o modelo GBT, considerado o estado da arte, e também foi superior ao próprio ChatGPT 4_{ZS}. No entanto, no *dataset* TA-Restaurantes, o AutoGluon apresentou desempenho inferior aos demais. Esses resultados indicam que as anotações feitas pelo ChatGPT podem ser usadas para treinar outros modelos e alcançar uma performance próxima à dele próprio ou mesmo superior. Mesmo com a possibilidade do ChatGPT 4 ter introduzido mais erros de anotações no *dataset*, ao treinar um modelo mais simples no AutoGluon, o modelo resultou em melhor generalização nos datasets Reli e Computer-BR, superando o próprio anotador.

Na Tabela 14, apresentam-se os resultados comparativos para a tarefa de identificação de polaridade (Tarefa 2). Pode-se notar que o desempenho do AutoGluon é inferior ao do ChatGPT 4_{ZS} em todos os conjuntos de dados examinados. Embora o desempenho do ChatGPT$_{ZS}$ tenha superado significativamente os padrões estabelecidos pelo modelo da literatura, a abordagem de utilizar o ChatGPT$_{ZS}$ como anotador automático para treinar o AutoGluon não funcionou tão bem quanto no caso anterior. É importante mencionar que o ChatGPT é baseado em um modelo muito grande e poderoso, treinado em uma vasta quantidade de dados textuais, então como seu desempenho foi muito superior nesta tarefa, um modelo mais simples treinado pelo AutoGluon não conseguiu superar seu desempenho. Apesar desse fato, os resultados do AutoGluon são melhores que o GBT em todos os casos, exceto para o TA-Restaurantes. Assim, pode-se concluir que o ChatGPT é uma potencial ferramenta de anotação útil em tarefas nas quais ele já apresenta um bom desempenho.

Na Tabela 15, apresentam-se os resultados comparativos para a Tarefa 3 (identificação de frases comparativas). O AutoGluon, treinado com anotações do ChatGPT 4_{ZS}, obteve um desempenho próximo a ele. Este resultado sugere que o AutoGluon conseguiu aprender efetivamente a partir das anotações fornecidas pelo modelo de linguagem. No entanto, seu desempenho foi inferior no conjunto de dados do Twitter, particularmente na pontuação F1, o que pode indicar que esse modelo teve dificuldades para generalizar em fontes de dados com conteúdo mais diversificado.

	Buscapé			**Twitter**		
	P	R	F_1	P	R	F_1
NB	**0.87**	**0.87**	**0.87**	**0.86**	**0.86**	**0.86**
ChatGPT 4	0.75	0.75	0.75	0.70	0.70	0.70
AutoGluon	0.75	0.71	0.70	0.72	0.57	0.51

Tabela 15: Identificação de sentenças comparativas (Tarefa 3) - usando ChatGPT$_{ZS}$ como anotador.

Uma possível explicação para o desempenho inferior do AutoGluon em relação ao ChatGPT pode estar relacionada às complexidades inerentes às arquiteturas dos modelos. Enquanto o ChatGPT foi extensivamente treinado em padrões linguísticos diversos e pode se adaptar a várias nuances dos dados, o AutoGluon pode

não extrapolar tão efetivamente a partir dos dados anotados apenas. Além disso, dados do Twitter, por serem mais informais e diversos, podem introduzir desafios adicionais que podem influenciar a capacidade de generalização de um modelo mais simples.

A partir dos resultados apresentados, pode-se deduzir que, mesmo com uma ligeira queda de desempenho, utilizar dados rotulados por um modelo de linguagem para treinar outros modelos de aprendizado de máquina continua sendo uma opção viável. Essa vantagem se torna particularmente evidente quando o modelo demonstra alto desempenho, como observado na análise de sentimentos de frases (Tarefa 2). Devido ao tamanho do ChatGPT 4, com estimados 1,7 trilhões de parâmetros, aproveitar suas capacidades para treinar modelos mais compactos, como o AutoGluon, pode fornecer uma vantagem significativa na implementação de soluções eficientes de aprendizado de máquina profundo.

6. Conclusões

Este artigo apresenta um estudo abrangente que investiga a eficácia de modelos de linguagem generativos em lidar com três tarefas relevantes de análise de sentimentos em português, utilizando diversos conjuntos de dados. Os resultados alcançados demonstram que os modelos ChatGPT, especialmente o GPT 4, podem ser utilizados com sucesso como modelos para identificação de polaridade em sentenças. Além disso, descobriu-se que o conjunto de dados anotado pelo ChatGPT pode ser usado para treinar modelos alternativos com impacto mínimo no desempenho, obtendo resultados comparáveis aos alcançados pelo próprio ChatGPT. Portanto, ele pode ser uma ferramenta útil quando tempo e custo são aspectos importantes na construção de modelos de aprendizado de máquina.

No entanto, para algumas outras tarefas, como identificação de subjetividade e sentenças comparativas, o ChatGPT não teve um bom desempenho como solução *zero-shot*. Sugere-se que isso ocorra devido a dois fatores: a dificuldade em construir prompts diretos e a menor ocorrência natural do assunto nos dados de treinamento do ChatGPT. Por exemplo, a identificação de sentimento em frases possui prompts mais precisos e é uma estrutura linguística muito comum em qualquer assunto textual, o que pode explicar o desempenho superior do ChatGPT nessa tarefa. Esse argumento se mostra plausível observando um melhor desempenho do MariTalk, treinado em português, especialmente na tarefa de identificação de subjetividade. De forma geral, observa-se que há um grande potencial de avanço em modelos de linguagem personalizados ao português, onde esperam-se melhores resultados conforme esses modelos ganharem capacidade e forem treinados em *datasets* maiores.

Em pesquisas futuras, há diversas vertentes a serem exploradas para aprimorar ainda mais os resultados. Uma área de foco será o aprimoramento das técnicas de engenharia de prompts para extrair resultados ainda melhores dos modelos testados. Além disso, planeja-se investigar o desempenho de outros modelos de linguagens generativas disponíveis na comunidade *Open Source*, expandindo a avaliação para abranger uma gama mais ampla de modelos e comparando sua eficácia em tarefas de análise de sentimento.

Agradecimentos

Este trabalho foi apoiado pelo Samsung Ocean Center, um programa de pesquisa e desenvolvimento da Universidade do Estado do Amazonas. Os autores também agradecem ao apoio financeiro da Fundação de Amparo à Pesquisa do Estado do Amazonas - FAPEAM através do Projeto NeuralBond (UNIVERSAL 2023 Proc. 01.02.016301.04300/2023-04).

Referências

Aires, João Paulo, Carlos Padilha, Christian Quevedo & Felipe Meneguzzi. 2018. A deep learning approach to classify aspect-level sentiment using small datasets. Em *International Joint Conference on Neural Networks (IJCNN)*, 1–8. 10.1109/IJCNN.2018.8489760

Almeida, Thales Sales, Hugo Abonizio, Rodrigo Nogueira & Ramon Pires. 2024. Sabiá-2: A new generation of Portuguese large language models. ArXiv:2403.09887 [cs.CL]. 10.48550/arXiv.2403.09887

de Araujo, Gladson, Tiago de Melo & Carlos Maurício S Figueiredo. 2024. Is ChatGPT an effective solver of sentiment analysis tasks in Portuguese? a preliminary study. Em *16th International Conference on Computational Processing of Portuguese*, 13–21.

Baeza-Yates, Ricardo & Berthier Ribeiro-Neto. 1999. *Modern information retrieval*. ACM Press, Addison-Wesley

Bahrini, Aram, Mohammadsadra Khamoshifar, Hossein Abbasimehr, Robert J. Riggs,

Maryam Esmaeili, Rastin Mastali Majdabadkohne & Morteza Pasehvar. 2023. ChatGPT: Applications, opportunities, and threats. Em *Systems and Information Engineering Design Symposium (SIEDS)*, 274–279. 10.1109/SIEDS58326.2023.10137850

Belal, Mohammad, James She & Simon Wong. 2023. Leveraging ChatGPT as text annotation tool for sentiment analysis. ArXiv [cs.CL]. 10.48550/arXiv.2306.17177

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford & Ilya Sutskever. 2020. Language models are few-shot learners. Em *Neural Information Processing Systems*, 1877–1901.

Carvalho, Caio Magno Aguiar, Hitoshi Nagano & Allan Kardec Barros. 2017. A comparative study for sentiment analysis on election Brazilian news. Em *11th Brazilian Symposium in Information and Human Language Technology*, 103–111.

Chang, Yupeng, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang & Xing Xie. 2023. A survey on evaluation of large language models. ArXiv [cs.CL]. 10.48550/arXiv.2307.03109

Christiano, Paul, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg & Dario Amodei. 2023. Deep reinforcement learning from human preferences. ArXiv [stat.ML]. 10.48550/arXiv.1706.03741

Ding, Bosheng, Chengwei Qin, Linlin Liu, Lidong Bing, Shafiq Joty & Boyang Li. 2022. Is GPT-3 a good data annotator? ArXiv [cs.CL]. 10.48550/arXiv.2212.10450

Elshawi, Radwa, Mohamed Maher & Sherif Sakr. 2019. Automated machine learning: State-of-the-art and open challenges. ArXiv [stat.ML]. 10.48550/arXiv.1906.02287

Erickson, Nick, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li & Alexander Smola. 2020. Autogluon-tabular: Robust and accurate automl for structured data. ArXiv [stat.ML]. 10.48550/arXiv.2003.06505

Fatouros, Georgios, John Soldatos, Kalliopi Kouroumali, Georgios Makridis & Dimosthenis Kyriazis. 2023. Transforming sentiment analysis in the financial domain with ChatGPT. *Machine Learning with Applications* 14. 100508. 10.1016/j.mlwa.2023.100508

Freitas, Cláudia, Eduardo Motta, R Milidiú & Juliana César. 2012. Vampiro que brilha... rá! desafios na anotaçao de opiniao em um corpus de resenhas de livros. *Encontro de Linguística de Corpus* 11. 22.

Friedman, Jerome H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* 1189–1232. 10.1214/aos/1013203451

Gilardi, Fabrizio, Meysam Alizadeh & Maël Kubli. 2023. ChatGPT outperforms crowd-workers for text-annotation tasks. Em *Proceedings of the National Academy of Sciences*, e2305016120. 10.1073/pnas.2305016120

Hayatin, Nur, Suraya Alias & Lai Po Hung. 2024. Trends and challenges in sentiment summarization: a systematic review of aspect extraction techniques. *Knowledge and Information Systems* 1–47. 10.1007/s10115-024-02075-w

Hou, Ziwei, Bahadorreza Ofoghi, Nayyar Zaidi & John Yearwood. 2024. Aspect-based fake news detection. Em *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 95–107. 10.1007/978-981-97-2266-2_8

Jiao, Wenxiang, Wenxuan Wang, Jen tse Huang, Xing Wang & Zhaopeng Tu. 2023. Is ChatGPT a good translator? Yes with GPT-4 as the engine. ArXiv [cs.CL]. 10.48550/arXiv.2301.08745

Kansaon, Daniel, Michele A Brandão, Julio CS Reis, Matheus Barbosa, Breno Matos & Fabrício Benevenuto. 2020. Mining Portuguese comparative sentences in online reviews. Em *Brazilian Symposium on Multimedia and the Web*, 333–340. 10.1145/3428658.3431081

Lazarini, Lucas, Fábio S Igarashi Anno, Eloize R Marques Seno & Helena M Caseli. 2023. Abordagens baseadas em léxicos para a classificação de sentimentos orientada aos alvos de opinião em comentários do domínio político. Em *XIV Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, 375–380. 10.5753/stil.2023.234206

Li, Rui, Guoyin Wang & Jiwei Li. 2023. Are human-generated demonstrations necessary for in-context learning? ArXiv [cs.LG]. 10.48550/arXiv.2309.14681

Liu, Bing. 2020. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi & Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys* 55(9). 1–35. 10.48550/arXiv.2107.13586

Lopes, Émerson, Ulisses Correa & Larissa Freitas. 2021. Exploring BERT for aspect extraction in Portuguese language. Em *The International FLAIRS Conference*, vol. 34, 10.32473/flairs.v34i1.128357

Loukas, Lefteris, Ilias Stogiannidis, Prodromos Malakasiotis & Stavros Vassos. 2023. Breaking the bank with ChatGPT: Few-shot text classification for finance. ArXiv [cs.CL]. 10.48550/arXiv.2308.14634

McCallum, Andrew & Kamal Nigam. 1998. A comparison of event models for Naive Bayes text classification. Em *Workshop on Learning for Text Categorization*, vol. 752 1, 41–48

de Melo, Tiago. 2022. SentiLexBR: An automatic methodology of building sentiment lexicons for the Portuguese language. *Journal of Information and Data Management* 13(3). 10.5753/jidm.2022.2504

de Melo, Tiago, Altigran da Silva & Edleno S de Moura. 2018. An aspect-driven method for enriching product catalogs with user opinions. *Journal of the Brazilian Computer Society* 24. 1–19. 10.1186/s13173-018-0080-4

Moraes, Silvia, André LL Santos, Matheus Redecker, Rackel M Machado & Felipe R Meneguzzi. 2016. Comparing approaches to subjectivity classification: A study on Portuguese tweets. Em *Conference on Computational Processing of the Portuguese Language*, 86–94. 10.1007/978-3-319-41552-9_8

de Oliveira, Miguel & Tiago de Melo. 2021. An empirical study of text features for identifying subjective sentences in Portuguese. Em *10th Brazilian Conference on Intelligent Systems*, 374–388. 10.1007/978-3-030-91699-2_26

Oliveira, Miguel V & Tiago de Melo. 2020. Investigating sets of linguistic features for two sentiment analysis tasks in Brazilian Portuguese web reviews. Em *XXVI Simpósio Brasileiro de Sistemas Multimídia e Web*, 45–48. 10.5753/webmedia_estendido.2020.13060

Pandey, Shubham V & AV Deorankar. 2019. A study of sentiment analysis task and it's challenges. Em *International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1–5. 10.1109/ICECCT.2019.8869160

Pires, Ramon, Hugo Abonizio, Thales Sales Almeida & Rodrigo Nogueira. 2023. Sabiá: Portuguese large language models. Em *Brazilian Conference on Intelligent Systems*, 226–240. 10.48550/arXiv.2304.07880

Qin, Chengwei, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga & Diyi Yang. 2023. Is ChatGPT a general-purpose natural language processing task solver? ArXiv [cs.CL]. 10.48550/arXiv.2302.06476

Rana, Toqir A & Yu-N Cheah. 2016. Aspect extraction in sentiment analysis: comparative analysis and survey. *Artificial Intelligence Review* 46(4). 459–483. 10.1007/s10462-016-9472-z

Rudolph, Jürgen, Samson Tan & Shannon Tan. 2023. ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? *Journal of applied learning and teaching* 6(1). 342–363. 10.37074/jalt.2023.6.1.9

Sallam, Malik, Nesreen Salim, Muna Barakat & Alaa Al-Tammemi. 2023. ChatGPT applications in medical, dental, pharmacy, and public health education: A descriptive study highlighting the advantages and limitations. *Narra J* 3(1). e103–e103. 10.52225/narra.v3i1.103

Schreiner, Maximilian. 2024. GPT-4 architecture, datasets, costs and more leaked. Acesso: 20-072024.

Seno, Eloize, Lucas Silva, Fábio Anno, Fabiano Rocha & Helena Caseli. 2024. Aspect-based sentiment analysis in comments on political debates in Portuguese: evaluating the potential of ChatGPT. Em *16th International Conference on Computational Processing of Portuguese*, 312–320

Stiilpen Junior, Milton & Luiz Henrique C Merschmann. 2016. A methodology to handle social media posts in Brazilian Portuguese for text mining applications. Em *22nd Brazilian Symposium on Multimedia and the Web*, 239–246. 10.1145/2976796.2976845

Tan, Kian Long, Chin Poo Lee & Kian Ming Lim. 2023. A survey of sentiment analysis: Approaches, datasets, and future research. *Applied Sciences* 13(7). 4550. 10.3390/app13074550

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser & Illia Polosukhin. 2023. Attention is all you need. ArXiv [cs.CL]. 10.48550/arXiv.1706.03762

Wankhade, Mayur, Annavarapu Chandra Sekhara Rao & Chaitanya Kulkarni. 2022. A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review* 55(7). 5731–5780. 10.1007/s10462-022-10144-1

Zhao, Biao, Weiqiang Jin, Javier Del Ser & Guang Yang. 2023. ChatAgri: Exploring potentials of ChatGPT on cross-linguistic agricultural text classification. *Neurocomputing* 557. 126708. 10.1016/j.neucom.2023.126708

Proposta recebida em Julho 2024 e aceite para publicação em Outubro 2024.

Detecção de posicionamentos a partir de textos usando expansão semiautomática de córpus

Stance detection from text using semi-automatic corpus expansion

Camila Farias Pena Pereira
Escola de Artes, Ciências e Humanidades - Universidade de São Paulo

Ivandré Paraboni
Escola de Artes, Ciências e Humanidades - Universidade de São Paulo

Resumo

A detecção computacional de posicionamentos—a tarefa de determinar, a partir de um texto de entrada, a atitude ou posição (e.g., favorável ou contrária) em relação a um tópico-alvo específico—geralmente depende de córpus rotulados com informações de posicionamentos para cada tópico de interesse. Uma vez que esses tópicos são em princípio ilimitados, a necessidade de novos córpus rotulados também o é. Como forma de amenizar algumas dessas dificuldades, este trabalho adapta para a detecção de posicionamento um método de expansão de córpus originalmente desenvolvido para a tarefa análoga de análise de sentimento. O método é então aplicado a um grande (46 mil instâncias) córpus de posicionamentos cobrindo seis temas de interesse político e/ou moral em português brasileiro, obtendo um aumento substancial no número de instâncias. Resultados de avaliação automática e humana sugerem que a adição de instâncias rotuladas semiautomaticamente ao conjunto de dados original não prejudica a precisão da classificação, e que os rótulos gerados automaticamente estão, em sua maioria, corretos.

Palavras chave

detecção de posicionamentos, expansão de córpus

Abstract

Computational stance detection—the task of determining, given an input text, the attitude (e.g., for or against) towards a particular target topic—usually makes use of annotated corpus as training data and, since possible topics are in principle unlimited, so is the need for new labelled datasets about every topic of interest. In order to overcome some of these challenges, the present work adapts to the stance prediction task an existing corpus expansion method that has been originally devised for sentiment analysis. The method is applied to a large (46K instances) Brazilian Portuguese corpus conveying stances towards six target topics of moral and/or political nature, achieving a substantial increase in the number of labelled instances. Results from both automatic and human evaluation suggest that adding semi-automatically labelled data to the corpus does not decrease accuracy, and that the majority of these labels are correct.

Keywords

stance detection, corpus expansion

1. Introdução

A detecção computacional de posicionamentos (Küçük & Can, 2020; Aldayel & Magdy, 2021) visa determinar, a partir de um texto fornecido como entrada, a atitude ou posição (por exemplo, favorável ou contrária) em relação a um determinado tópico-alvo de interesse. A tarefa permite identificar, por exemplo, se um indivíduo ou grupo está concordando ou discordando de uma determinada afirmação, se está assumindo uma determinada posição sobre um tópico possivelmente controverso ou agressivo (da Silva et al., 2020) ou, de forma mais geral, estimar qual o reflexo de uma afirmação presente no texto sobre o alvo (por exemplo, sendo a favor ou contra ele). Este último caso, também conhecido como detecção de posicionamentos orientada a alvos (do inglês, *target-based stance detection*), é o foco do presente trabalho.

A detecção de posicionamentos geralmente assume a forma de uma tarefa de aprendizado de máquina supervisionada baseada em córpus anotados (por exemplo, postagens em redes sociais rotuladas manualmente com posições contrárias/favoráveis a um alvo específico) e é, em princípio, análoga à análise de sentimento (AS), ou seja, a tarefa de determinar o sentimento positivo/negativo no texto (Zhang & v Wang, 2018).

DOI: 10.21814/lm.16.2.436
This work is Licensed under a Creative Commons Attribution 4.0 License

No entanto, AS é uma tarefa de PLN possivelmente mais superficial no sentido de que os modelos de AS podem, em princípio, usar qualquer domínio suficientemente próximo (por exemplo, resenhas de filmes como em '*o filme foi terrível*') como dados de treinamento para então inferir sentimentos em outros domínios (por exemplo, análises de produtos, como em '*a bateria do aparelho é péssima*'). A tarefa de detecção de posicionamentos, por outro lado, depende muito mais de conhecimento semântico associado ao tópico-alvo, fazendo-se necessário criar um novo conjunto de treinamento específico para cada alvo de interesse. Considere-se os seguintes exemplos:

(1) *Claro que o certo é tomar cloroquina. Você ainda pode morrer de Covid-19, mas nunca de malária!*

(2) *Se a vacina Coronavac é tão eficaz, porque é que nenhum país Europeu a utiliza?*

Ambos os exemplos expressam um posicionamento contrário a um medicamento ou tratamento que era discutido no contexto da pandemia de Covid-19. No entanto, além de misturar termos positivos (por exemplo, "certo") e negativos (por exemplo, "morrer"), notamos que essas duas afirmações têm pouco em comum, por exemplo, em termos de vocabulário ou estrutura. Como resultado, um córpus de treinamento de posicionamento em relação a um tópico pode não ser necessariamente útil à construção de um modelo de detecção de posicionamento em relação a outro.

Uma possível estratégia para amenizar a escassez de dados rotulados seria a expansão de córpus de posicionamentos a partir de dados rotulados já existentes. De forma mais específica, o presente trabalho aborda uma estratégia de expansão de córpus originalmente desenvolvida para a tarefa de análise de sentimento por Brum & das Graças Volpe Nunes (2018), aqui adaptada para detecção de posicionamento na língua portuguesa. Esta estratégia foi aplicada a um experimento de expansão de córpus de posicionamentos, e resultados de avaliação intrínseca e humana são relatados.

O restante deste artigo está organizado da seguinte forma. Após uma visão geral dos trabalhos existentes na área de detecção de posicionamentos na seção 2, o presente trabalho é dividido em três partes principais. A seção 3 descreve o córpus de posicionamento a ser expandido automaticamente. A seção 4 apresenta os modelos classificadores a serem tomados como base para o método de expansão, seus resultados individuais e sua análise de características de aprendizado. Na seção 5 nossa atenção se volta para o método de expansão do córpus propriamente dito, descrevendo sua arquitetura e seus resultados. Por fim, a seção 6 apresenta nossas principais conclusões e sugestões de trabalhos futuros.

2. Trabalhos relacionados

A Tabela 1 apresenta uma visão geral dos estudos sobre a detecção computacional de posicionamentos, categorizados de acordo com o gênero de texto, idioma, modelo de representação textual e método computacional empregado. Detalhes adicionais são discutidos a seguir.

A partir deste levantamento, é possível constatar que a grande maioria dos estudos da área tende a ser baseada na rede social Twitter/X. Isso talvez seja explicável pela maior facilidade de coleta de dados que esta plataforma costumava oferecer. Além disso, assim como em outros tópicos de pesquisa em PLN, estudos de detecção de posicionamentos tendem a ser voltados ao idioma inglês, na maioria dos casos enfocando o uso de poucos conjuntos de dados tidos como padrão da área, como o córpus SemEval-2016 (Mohammad et al., 2016), VAST (Allaway & McKeown, 2020), e P-Stance (Li et al., 2021).

Com relação aos modelos de representação textual, é possível observar que os estudos da área seguem a evolução natural dos métodos empregados em outras áreas do PLN. Em estudos mais antigos, isso é sugerido pelo maior uso de modelos de contagens de *tokens* do tipo *bag-of-words* ou equivalente, frequentemente combinados com métodos de classificação tradicionais do tipo regressão logística ou SVMs. Ao longo dos anos, estas abordagens foram sendo substituídas pelo uso de *word embeddings* estáticos e posteriormente BERT, com uso crescente de métodos de aprendizado neural.

Em complemento ao levantamento das abordagens de detecção de posicionamentos existentes, foi realizado também um levantamento dos conjuntos de dados disponíveis considerando-se apenas a versão da tarefa orientada a alvos, que é o foco do presente trabalho. Os estudos deste tipo são sumarizados na Tabela 2.

3. O córpus UstanceBR r2

O presente estudo faz uso do córpus de posicionamentos *UstanceBR* versão r2, apresentado em Pereira et al. (2023). O córpus é uma coleção de *tweets* em língua portuguesa que foram manualmente anotados com informação referente ao

Estudo	Gênero	Idioma	Representação textual	Método
Zhou et al. (2019)	T	En	WE	CNN
Tun & Hninn Myint (2019)	T	En	BoW	NB
Siddiqua et al. (2019)	T	En	WE	LSTM
Sane et al. (2019)	T	Hi	BoW,WE	CNN
Li & Caragea (2019)	T	En	WE,l	LSTM
Fang et al. (2019)	C	En	BERT	TE
Mayfield & Black (2019)	D	En	WE,BERT	LR
dos Santos & Paraboni (2019)	T	Pt	BoW	NB, LR, RN
Xu et al. (2019a)	T	En	WE	RNN
Hassan & Lee (2019)	C	En	BoW	RN
Xu et al. (2019b)	D	En	WE	CNN,LSTM
Hanawa et al. (2019)	W	En	WE	LSTM
Bugueño & Mendoza (2019)	T	En	BERT	CNN
Schaefer & Stede (2019)	T	En	BoW,WE	SVM
Popat et al. (2019)	D	En	BERT	RN
D'Andrea et al. (2019)	D	It	BoW	SVM
Pavan et al. (2020)	T	Pt	WE	LSTM
Kovačević et al. (2020)	C	Se	l	-
Carvalho et al. (2020)	T	En	WE	LSTM
Conforti et al. (2021)	T,N	En	BERT	Softmax
Ayyub et al. (2021)	T	En	WE,PoS,l	NB,SVM,CNN
Su et al. (2021)	T	Ch	WE	Softmax
Samih & Darwish (2021)	T	En	BERT	SVM,Softmax
Zhu et al. (2021)	C	En	BERT	Softmax
Flores et al. (2022)	E	Pt	BERT	LSTM
Zhao et al. (2022)	T	En	BERT	Softmax
Pavan & Paraboni (2022)	T	Pt	BERT	LSTM
Ng & Carley (2022)	T	En	BERT	Softmax
Pavan et al. (2023)	C	Pt	BERT	LSTM
Chunling et al. (2023)	T	En	BERT	GAN
Zhang et al. (2023a)	T	En	none	ChatGPT
Wen & Hauptmann (2023)	N	En	BART	RN
Zhang et al. (2023b)	T,N	En	RoBERTa	RN
Cavalheiro et al. (2023)	T	Pt	BERT,BoW	BiLSTM,EN
Li et al. (2023)	T,N	En	BERT	ChatGPT
de Sousa & Becker (2023)	T	En	BoW	LR,SVM,RF
Zhang et al. (2024)	T,N	En	BERT	RN

Tabela 1: Estudos de detecção computacional de posicionamentos de acordo com gênero (A = avaliações; C = textos ou redações; D = debates Online; E = respostas à solicitações online; T = Twitter; N = notícias; ; W = Wikipedia, idioma (Ch = Chinês; En = Inglês; Hi = Hindi; It = Italiano; Pt = Português; Se = Sérvio), modelos de representação textual (BERT = BERT (Devlin et al., 2019); BoW = *bag-of-words*; D = Dependências Sintáticas; EL = características extralinguísticas; l = léxicos; PoS = *part-of-speech*; WE = Word Embeddings); e método computacional (CNN = Redes Neurais Convolucionais; EN = *ensemble*; GAN = redes neurais adversárias; ILP = *Inductive Logic Programming*; LSTM = *Long Short-term Memory networks*; RF = *Random Forest*; RL = Regressão Logística; NB = Naive Bayes; SVM = *Support Vector Machines*; RN = outros tipos de redes neurais).

posicionamento dos autores quanto a alvos de natureza altamente polarizada. Uma versão prévia deste córpus, chamada *UstanceBR* r1, foi apresentada em Pavan & Paraboni (2022) e não será aqui discutida.

O córpus *UstanceBR* r2 é composto de 86,8 mil *tweets*, totalizando cerca de 1,7 milhões de palavras, rotulados com informação de posicionamento binário (contra ou a favor) em relação a seis tópicos: presidentes brasileiros (Lula e Bolsonaro), medicamentos discutidos durante o período da pandemia Covid-19 (a vacina Coronavac e a Cloroquina), e instituições (a rede de TV Globo e a igreja). Os posicionamentos rotulados são aproximadamente balanceados por classe, e somam-se a um conjunto menor classificado como 'Outros' que contempla *tweets* para os quais não foi obtida concordância entre anotadores, ou que não apresentavam um posicionamento claro em relação ao alvo.

Estudo	Gênero	Idioma	Instâncias (mil)	Rotulação
Mohammad et al. (2016)	T	En	4.2	t
Magdy et al. (2016)	T	En	336.3	p
Taulé et al. (2017)	T	Ca,Sp	10.8	t
Darwish et al. (2017)	T	Ar	33.0	t
Sobhani et al. (2017)	T	En	4.5	t
Conforti et al. (2020)	T	En	51.3	t
Pavan et al. (2023)	C	Pt	4.1	t,u
Mutlu et al. (2020)	T	En	14.4	t
Allaway & McKeown (2020)	O	En	23.6	t
Lai et al. (2020)	T	En,Fr,It,Sp,Ca	14.4	t
Glandt et al. (2021)	T	En	6.1	t
Jaziriyan et al. (2021)	T	Ar	9.6	t
Geiss et al. (2022)	R	En	2,717	u
Chen et al. (2022)	T	En,Fr,De,Du,Sp,*	17.9	t
Pereira et al. (2023)	T	Pt	86.8	t,u

Tabela 2: Córpus para detecção de posicionamentos orientada a alvo de acordo com gênero (T = Twitter ; R = Reddit ; C = textos ou redações ; O = opiniões), idioma (Ar=Árabe, Ca=Catalão, De=Alemão, Du=Holandês, En=Inglês, Fr=Francês, It=Italiano, Pt=Português, Sp=Espanhol, *=Outros), número de instâncias, e método de rotulação dos dados (t= em nível textual; p = propagação de rótulos ; u=em nível de usuário ou *timeline*).

A Tabela 3 apresenta estatísticas descritivas do conjunto de posicionamentos rotulados do córpus.

Para os experimentos descritos nas próximas seções, foram utilizados apenas os *tweets* com rótulos contra e a favor, com a divisão treino/teste denominada r2 em Pereira et al. (2023), que corresponde a 75% para treino e 25% para teste.

4. Detecção de posicionamentos

Como forma de avaliar o grau de dificuldade da tarefa de detecção de posicionamentos no córpus *UstanceBR*, e também estabelecer resultados de *baseline* a serem considerados na tarefa de expansão automática do córpus a ser discutida na próxima seção, foram conduzidos experimentos de classificação binária para os seis tópicos-alvo existentes.

Os experimentos conduzidos consideram o uso de três tipos de classificadores de posicionamentos que, dado um tópico-alvo de interesse, objetivavam atribuir um rótulo 'a favor' ou 'contrário' a cada *tweet* fornecido como entrada. Dois destes classificadores são do tipo regressão logística baseada em um modelo de representação textual *bag-of-words* utilizando contagem de palavras ou de caracteres, aqui denominados *RegLog.word* e *RegLog.char*, respectivamente. O terceiro classificador é baseado em um modelo pré-treinado de língua do tipo BERT (Devlin et al., 2019), que será aqui referido como *BERT*. O código destas implementações é disponibilizado publicamente para reuso[1].

Para os classificadores *RegLog.word* e *RegLog.char*, os seguintes parâmetros foram utilizados de forma padronizada para todos os tópicos-alvo: regularização com penalidade L_2, tolerância para o critério de parada igual a $1e^{-1}$, número máximo de iterações igual a 100, e pesos das classes igual a 1. A modelagem de características é baseada na contagem Tf-Idf de palavras ou n-gramas de caracteres de tamanho 2 a 3, a depender do modelo em questão.

Para o classificador *BERT*, utilizou-se *BERTabaporu* (da Costa et al., 2023), um modelo BERT pré-treinado em um conjunto de 237 milhões de *tweets* em língua Portuguesa, entre os quais estão incluídos os textos de treino do córpus *UstanceBR*. Para treinamento e classificação, foi usada a classe *ClassificationModel* da biblioteca *Simple Transformers*[2] com as opções '*reprocess_input_data*' e '*overwrite_output_dir*' definidas como *"true"*. O treinamento foi feito em uma época, pois em testes preliminares não foi verificado um ganho de medida F_1 que justificasse o uso de um tempo maior. Para todos os demais parâmetros, foi usada a opção *default*, que inclui a definição de um *batch size*

[1] https://github.com/camilafpp/expansaoUstanceBR

[2] https://simpletransformers.ai/docs/classification-models/

Alvo	Contra	A favor	Outros	Total	Palavras	Palavras/*Tweets*
Bolsonaro	5.565	3.848	1.668	11.082	259.521	23,42
Lula	4.514	3.806	13.934	22.254	422.064	18,97
Coronavac	4.058	3.915	932	8.905	252.663	28,37
Cloroquina	3.978	4.017	1.785	9.780	277.824	28,41
Globo	3.341	2.672	15.253	21.266	214.876	10,10
Igreja	3.539	3.598	6.329	13.466	322.289	23,93
Total	24.995	21.857	39.901	86.753	1.749.237	20,70

Tabela 3: Estatísticas descritivas do córpus UstanceBR r2 em Pereira et al. (2023).

	RegLog.word		**RegLog.char**		**BERT**	
Alvo	Acurácia	F_1	Acurácia	F_1	Acurácia	F_1
Bolsonaro	0,77	0,75	0,79	0,77	0,84	**0,80**
Lula	0,76	0,76	0,79	**0,79**	0,79	0,76
Coronavac	0,77	0,77	0,78	0,78	0,80	**0,81**
Cloroquina	0,80	0,80	0,80	0,80	0,80	**0,80**
Globo	0,79	0,78	0,81	0,81	0,85	**0,81**
Igreja	0,77	0,77	0,78	0,78	0,84	**0,84**

Tabela 4: Resultados da detecção de posicionamentos no córpus UstanceBR r2.

de 8 e o comprimento de sequência máximo de 128. A função *softmax* é acoplada à saída do modelo para obtenção dos rótulos das predições.

4.1. Avaliação da tarefa de detecção de posicionamentos

Para fins de avaliação dos modelos descritos na seção anterior, foi utilizada a divisão padrão dos textos do córpus *UstanceBR* r2 (Pereira et al., 2023). Com base nesta divisão, os modelos propostos foram treinados e testados individualmente para cada um dos seis tópicos-alvo do córpus. Os resultados obtidos são sumarizados na Tabela 4, em que os melhores escores de medida F_1 para cada alvo estão destacados.

Os resultados apresentados na Tabela 4 indicam que o modelo *BERT* obteve os melhores resultados de classificação de forma geral, seguido pelo *RegLog.char*. Essa variação, no entanto, foi diferente para cada tópico, apresentando uma diferença de 7 pontos percentuais no caso do tópico 'Igreja' e de apenas um ponto no caso do alvo 'Cloroquina'.

4.2. Cômputo da importância das características de aprendizado

Para melhor entender o comportamento dos modelos de classificação desenvolvidos, foi conduzida uma análise complementar dos termos considerados mais relevantes para cada classe, tomando-se por base para este fim o modelo *RegLog.word*. A escolha desse modelo para a análise foi motivada pela observação de que *RegLog.char* e *BERT* são baseados em características de interpretação humana mais difícil.

A análise do modelo *RegLog.word* fez uso da biblioteca eli5[3], que fornece uma explicação dos resultados do modelo com base nas suas características mais importantes, ou seja, calculando o peso relativo de cada *token* na classificação. Esses pesos, que são calculados a partir de permutações de características (Breiman, 2001), indicam a mudança no resultado da métrica de avaliação quando a característica em questão é substituída por outra. Os pesos assim computados são positivos para posicionamentos favoráveis e negativos para posicionamentos contrários, devendo-se considerar o seu valor absoluto para estimar sua importância. Os resultados obtidos são sumarizados na Tabela 5.

Com base nos resultados da Tabela 5, observa-se que as características mais importantes são, de forma geral, termos intuitivamente consistentes com os alvos dos posicionamentos, e não termos de pouco conteúdo semântico como pronomes ou preposições, o que poderia levantar suspeitas de *overfitting*. Nota-se também que as características mais importantes pertencem, de modo geral, a um vocabulário específico associado a cada tópico, e na maioria dos casos não correspondem a termos de caráter mais genérico

[3] https://eli5.readthedocs.io/en/latest/autodocs/eli5.html#eli5.explain_weights

Lula		Coronavac		Globo		Bolsonaro		Cloroquina		Igreja	
peso	caract.	peso	caract.	peso	caract.	peso	caract.	peso	caract.	peso	caract.
3,586	presidente	3,449	gado	3,464	amo	5,307	presidente	3,996	anos	2,989	vou
2,877	moro	3,351	bolsonaro	3,270	na	3,760	nosso	3,930	hidroxicloroq.	2,527	ir
2,820	contra	3,021	doses	2,725	parabéns	3,360	mídia	3,923	china	2,506	saudade
2,602	provas	2,769	butantan	2,507	série	2,875	está	3,805	vidas	2,409	nossa
2,509	golpe	2,352	coronavac	2,487	filme	2,741	imprensa	3,698	esquerda	2,259	amanhã
2,390	livre	2,165	bozo	2,369	obrigada	2,724	esquerda	3,425	governadores	2,185	hoje
2,308	coração	2,042	mil	2,251	novela	2,710	parabéns	3,271	chinês	1,999	maria
2,220	lula	1,984	vacinas	2,040	passando	2,642	povo	3,118	globo	1,952	fui
2,029	perseguição	1,833	gente	2,008	plantão	2,633	stf	3,101	azitromicina	1,857	indo
1,942	julgamento	1,822	instituto	1,771	bbb	2,595	apoio	2,803	uip	1,843	senhor
-4,582	bandido	-3,488	vachina	-5,170	lixo	-3,861	via	-3,561	gado	-3,571	universal
-4,227	corrupção	-3,173	dória	-4,479	globolixo	-2,999	cu	-3,234	cloroquina	-3,107	evangélica
-4,025	ladrão	-2,852	contrato	-3,099	merda	-2,734	eu	-3,205	trump	-3,039	dinheiro
-3,409	condenado	-2,809	dinheiro	-2,247	seus	-2,566	merda	-2,772	bozo	-2,337	igrejas
-3,226	silva	-2,803	chinesa	-2,091	vergonha	-2,252	tá	-2,350	estoque	-2,314	eles
-3,088	cadeia	-2,658	doria	-2,084	tomar	-2,065	minha	-2,241	capitão	-2,179	católica
-3,061	presidiário	-2,574	china	-1,974	se	-2,000	coronavírus	-2,170	ozônio	-2,003	crente
-3,052	corrupto	-2,318	obrigar	-1,945	não	-1,968	ta	-2,119	gripezinha	-1,887	estado
-2,102	criminoso	-2,067	cobaia	-1,891	contra	-1,900	votou	-2,033	ah	-1,881	religião
-2,047	babaca	-1,965	essa	-1,782	bosta	-1,770	falando	-2,021	eua	-1,746	inferno

Tabela 5: Características mais importantes no modelo *RegLog.word*.

que poderiam ser relacionados a mais de um posicionamentos ou tópico.

Além de os classificadores exibirem um comportamento razoavelmente próximo do esperado, observa-se ainda que a maioria das características corresponde a palavras de conteúdo, como 'presidente', 'doses', 'filme', etc., e que não podem ser classificadas como representando um sentimento positivo ou negativo na tarefa de análise de sentimentos. Isso reforça o entendimento da natureza tradicional da tarefa de classificação de posicionamento como distinta da de classificação de sentimento, sendo possivelmente mais dependente de conhecimento semântico (Küçük & Can, 2020).

5. Expansão de dados

Apesar dos resultados de classificação de modo geral positivos reportados na seção anterior, observa-se que a massa de dados coletada nas etapas iniciais de construção do córpus *UstanceBR* permanece em grande parte sem uso, dado que apenas uma pequena porção (aproximadamente 24%) dos dados foi anotada por razões de custo. A Tabela 6 apresenta estatísticas descritivas do conjunto de treino e testes rotulado manualmente, e o número de instâncias não rotuladas de cada subconjunto (alvo).

O aproveitamento de um volume maior de dados, desde que rotulados de forma confiável, poderia viabilizar a construção de modelos mais

Alvo	Treino	Teste	Não Rotulado
Bolsonaro	7.060	2.354	31.995
Lula	6.240	2.080	27.489
Coronavac	5.979	1.994	35.790
Cloroquina	5.996	1.999	35.181
Globo	4.509	1.504	29.854
Igreja	5.352	1.785	34.590

Tabela 6: Instâncias de treino e teste rotulados manualmente e porção não rotulada do córpus *UstanceBR* r2.

robustos para a tarefa de classificação discutida na seção anterior. Assim, consideramos uma estratégia de expansão semiautomática do conjunto de dados existente usando os modelos *RegLog.word*, *RegLog.char* e *BERT* descritos na seção anterior. O termo 'semiautomática' denota aqui a combinação do processo de anotação manual inicial com a presente automação.

Para a expansão do córpus, foi escolhido o método *self-training* de aprendizado semi-supervisionado (Zhu, 2005). Um procedimento similar foi utilizado em Brum & das Graças Volpe Nunes (2018) para classificação de sentimento, utilizando o método de *self-training* para expandir um conjunto de dados de texto rotulado manualmente em um *framework* denominado CasSUL. Uma das premissas deste método é a de que tarefas com um certo grau de subjetividade, como seria o caso da própria análise de

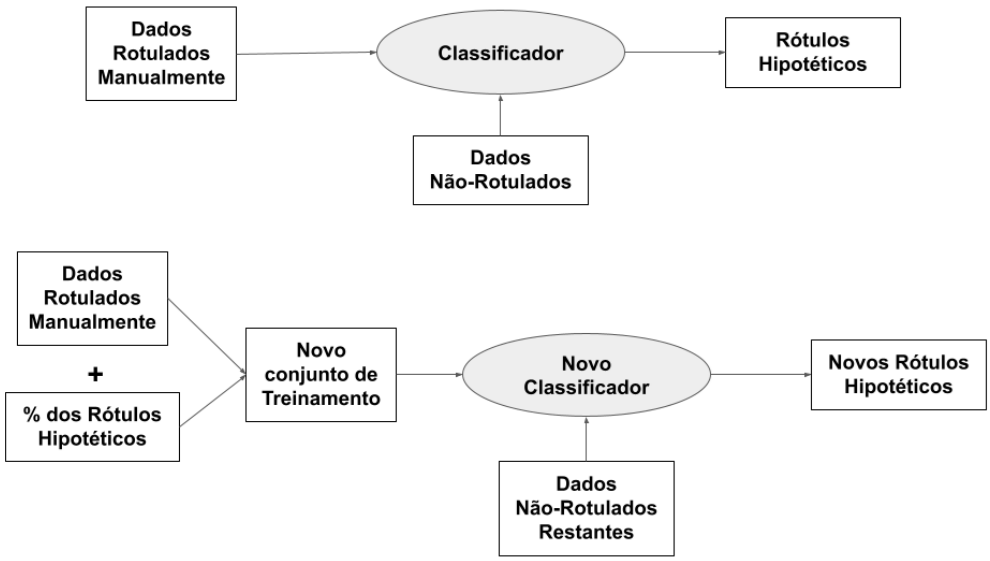


Figura 1: Modelo de expansão do conjunto de dados.

sentimentos, não podem prescindir de anotação manual como uma parte do processo.

Em Brum & das Graças Volpe Nunes (2018), foram conduzidos experimentos utilizando seis classificadores: *Support Vector Machines* (SVM), *Bernoulli Naive Bayes*, Regressão Logística, *Multilayer Perceptron* (MLP), Árvores de Decisão e *Random Forest*. Para todas as alternativas exceto Regressão Logística, foi realizada uma busca em grade para definir os melhores parâmetros de cada modelo. Além disso, foram realizados experimentos com diversas combinações de características para além do texto puro, como uso de *emoticons*, *emoji* e *part-of-speech*. Os melhores resultados foram encontrados utilizando-se classificadores do tipo MLP e Regressão Logística, porém sem ultrapassar significantemente o resultado obtido com o conjunto rotulado manualmente. No entanto, o *framework* CasSUL permitiu expandir o conjunto de dados original em 8 vezes.

5.1. Procedimento de expansão

A Figura 1 apresenta uma representação gráfica do processo de expansão de córpus proposto. Na primeira parte, os dados anotados manualmente são utilizados para treinar um classificador utilizado para rotular dados de forma dita hipotética. A parte inferior ilustra como um percentual desses dados é adicionado ao conjunto rotulado manualmente, gerando um novo conjunto de treinamento. Um classificador é então novamente treinado com esses dados e o ciclo se repete.

De maneira análoga ao *framework* CasSUL, a expansão do conjunto de dados de posicionamentos também se inicia a partir dos modelos de classificação treinados com o conjunto rotulado manualmente. Esses modelos são então usados para classificar as instâncias restantes que, estando assim rotuladas de forma hipotética, são avaliadas para possível agregação ao córpus. Para esta avaliação de elegibilidade, as novas instâncias são ordenadas por confiabilidade, aqui definida como a probabilidade da predição fornecida pelo classificador. A qualidade do conjunto resultante após a adição de novos dados é testada treinando-se um novo modelo, e então avaliando-se seu desempenho com base na medida F_1 no mesmo conjunto de teste. O processo então se repete, com o novo classificador sendo usado para rotular os dados que não foram selecionados, até que não seja mais possível fazer novas adições dentro do limite dado pelo valor de *threshold* em uso.

Apesar de fazer uso do mesmo método adotado no *framework* CasSUL, o presente modelo apresenta algumas diferenças fundamentais em relação a Brum & das Graças Volpe Nunes (2018). Em primeiro lugar, observa-se que o presente trabalho trata de uma tarefa distinta da classificação de sentimento, que é a detecção de posicionamento em relação a um determinado alvo. Conforme destacado em Küçük & Can (2020), sentimento e posicionamento não estão necessariamente correlacionados. Para exemplificar, considere-se os dois textos abaixo, ambos expressando um posicionamento favorável ao alvo 'Igreja', porém com sentimentos opostos (ou seja, negativo no primeiro caso e positivo no segundo).

Alvo	RegLog.word *Threshold*	RegLog.word Iterações	RegLog.char *Threshold*	RegLog.char Iterações	BERT *Threshold*	BERT Iterações
Bolsonaro	1%	11	1%	16	1%	1
Lula	1%	13	10%	2	1%	1
Coronavac	1%	8	5%	2	1%	3
Cloroquina	1%	5	1%	2	1%	1
Globo	1%	6	1%	1	1%	3
Igreja	1%	14	5%	3	1%	5

Tabela 7: Parâmetros ótimos de cada modelo.

'*Eu vou na igreja para me proteger de todo esse mal.*'

'*Fui na igreja hoje, me senti muito bem.*'

Uma segunda diferença do presente estudo em relação ao de Brum & das Graças Volpe Nunes (2018) é a estratégia de modelagem de dados adotada, no presente caso utilizando não apenas a abordagem *bag-of-words* (BoW) tradicional, mas também o uso de *transformers* do tipo BERT (Devlin et al., 2019). Além disso, o presente trabalho enfoca apenas a construção de modelos puramente textuais, não contemplando o uso de características como *emoji*, *emoticons* e etiquetas de *part of speech* que foram utilizadas em Brum & das Graças Volpe Nunes (2018), e que apresentaram resultados inferiores ao de modelos textuais.

Finalmente, outra diferença significativa entre o presente trabalho e o *framework* CasSUL é ainda a questão do balanceamento dos rótulos gerados. Para evitar o problema de desbalanceamento de classes apontado em Brum & das Graças Volpe Nunes (2018), ao invés de fazer a seleção apenas com base na confiabilidade do classificador, no presente trabalho são selecionadas as N instâncias mais confiáveis de cada classe, onde N representa a metade do volume de dados a ser selecionado com base no *threshold* definido. Assim, o conjunto de dados permanece balanceado durante todo o procedimento, evitando que um desbalanceamento de classes viesse a ser propagado para as etapas seguintes de forma cumulativa.

Para determinar o número de instâncias a serem selecionadas, foram experimentadas diferentes proporções do conjunto não rotulado inicial considerando-se um valor de *threshold* de 1%, 5%, 10%, 25% e 40%. No caso do modelo *BERT*, entretanto, dado o custo computacional exigido para treino de múltiplas configurações de *threshold*, os experimentos foram limitados à adição de 1% do conjunto não-rotulado e até 5 iterações em todos os conjuntos.

A Tabela 7 sumariza os parâmetros ótimos de cada modelo referentes ao valor de *threshold* e número de interações empregados.

5.2. Resultados da expansão

A Figura 2 ilustra a evolução dos resultados do classificador *RegLog.word* com diferentes números de iterações e valores de *threshold*.

No caso do modelo *RegLog.word*, observa-se que a adição sucessiva de 1% dos dados rotulados alcançou o melhor desempenho, com ganho de até 3% na medida F_1. Além disso, em todos os tópicos esse desempenho atinge seu máximo com a adição de menos de 20% dos dados não-rotulados, tendendo a diminuir com a incorporação de maiores volumes.

Já no caso do modelo *RegLog.char*, ilustrado na Figura 3, os resultados obtidos foram ligeiramente superiores. No entanto, a diferença entre o desempenho do modelo treinado com o conjunto de dados rotulado manualmente e o conjunto expandido foi modesta, com ganho máximo de 2% de medida F_1.

Finalmente, a Figura 4 apresenta os resultados da expansão usando o modelo *BERT* com adição de uma porção fixa de 1% do conjunto não-rotulado, em que a interação '0' diz respeito ao resultado obtido pelo conjunto rotulado manualmente. Como discutido, no caso deste modelo específico foi realizado um número menor de interações devido ao seu custo computacional, mas ainda assim foi possível observar um comportamento similar ao dos modelos *RegLog.word* e o *RegLog.char*. Este resultado sugere que não seriam obtidos resultados significativamente superiores aos atuais caso o procedimento fosse ampliado.

5.2.1. Avaliação Intrínseca

Como uma primeira forma de avaliar a qualidade dos rótulos atribuídos de forma semiautomática, foram conduzidos experimentos de classificação

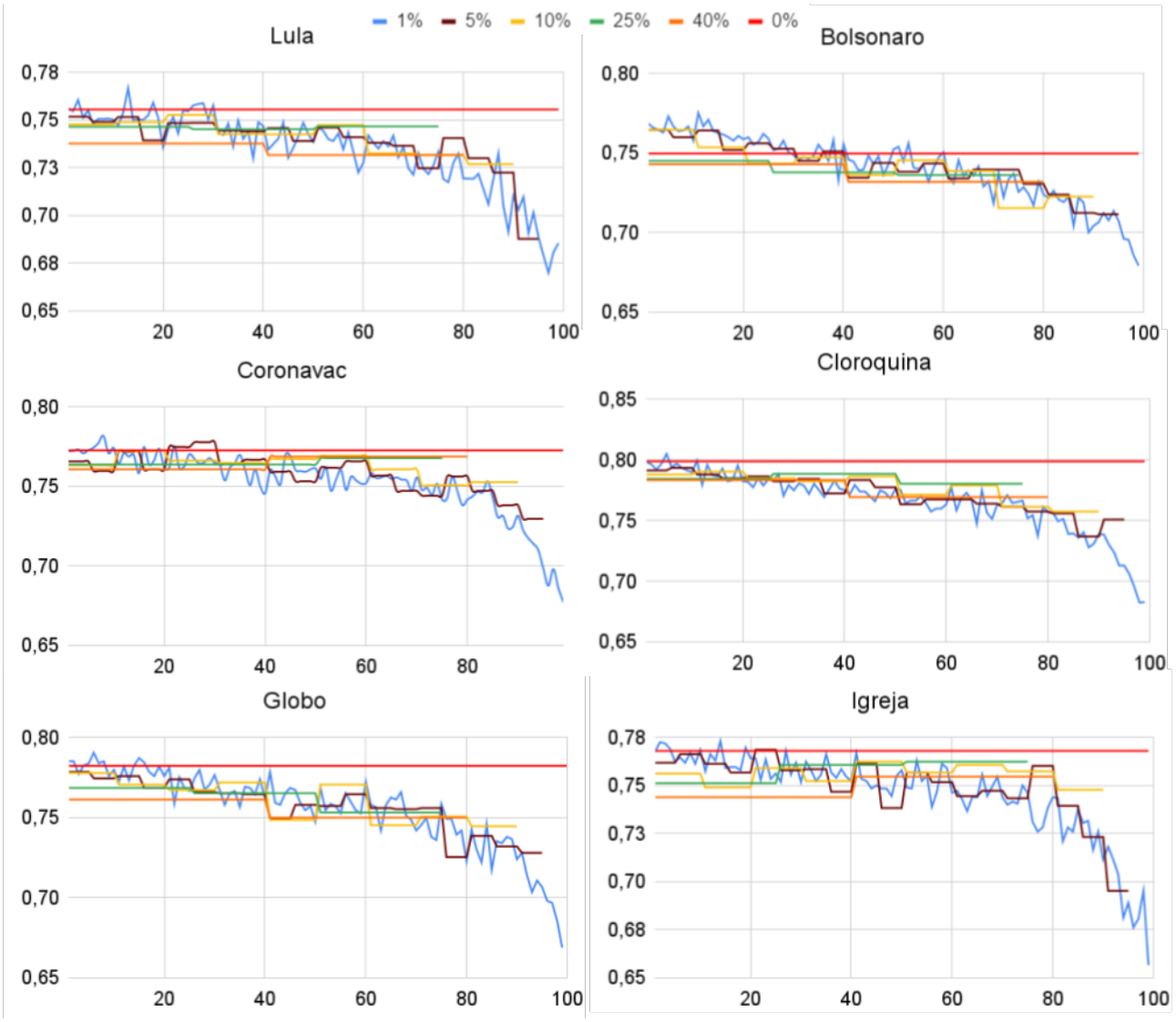


Figura 2: Expansão usando RegLog.word (horizontal: número de iterações; vertical: medida F_1).

similares aos relatados na seção 4.1, utilizando-se o córpus expandido como conjunto de treinamento para a tarefa de detecção de posicionamentos. O objetivo desses experimentos foi o de comparar o resultado da predição de posicionamento com e sem dados expandidos, tomando-se por base o mesmo conjunto de teste fixo.

A Tabela 8 sumariza os resultados obtidos com uso de cada um dos três modelo para cada os seis alvos do córpus. Os resultados originais do modelo *BERT*, que são os mesmos já reportados na Tabela 4 da seção 4.1, são aqui reproduzidos apenas para comparação com os resultados da expansão.

Os resultados apresentados na Tabela 8 sugerem que a inclusão de instâncias rotuladas de forma semiautomática não prejudicou o desempenho dos classificadores. Na verdade, algumas classes mostram inclusive um aumento discreto em medida F_1 médio, embora os modelos originais já tenham sido otimizados para o conjunto de dados atual. Este resultado é consistente com os obtidos pelo *framework* de expansão de córpus de análise de sentimentos CasSUL em Brum & das Graças Volpe Nunes (2018).

5.2.2. Avaliação Humana

Em complemento à avaliação intrínseca descrita na seção anterior, foi conduzida também uma avaliação humana dos dados rotulados de forma semiautomática utilizando-se o modelo *BERT* como referência. Para este fim, três juízes foram solicitados a rotular manualmente um conjunto de *tweets* aleatoriamente selecionados a partir de cada um dos melhores conjuntos de cada tópico usando as opções 'contrário', 'favorável' ou 'outro'. Embora os dados expandidos a serem avaliados fossem totalmente rotulados com informação de classe binária, esta terceira opção foi ofere-

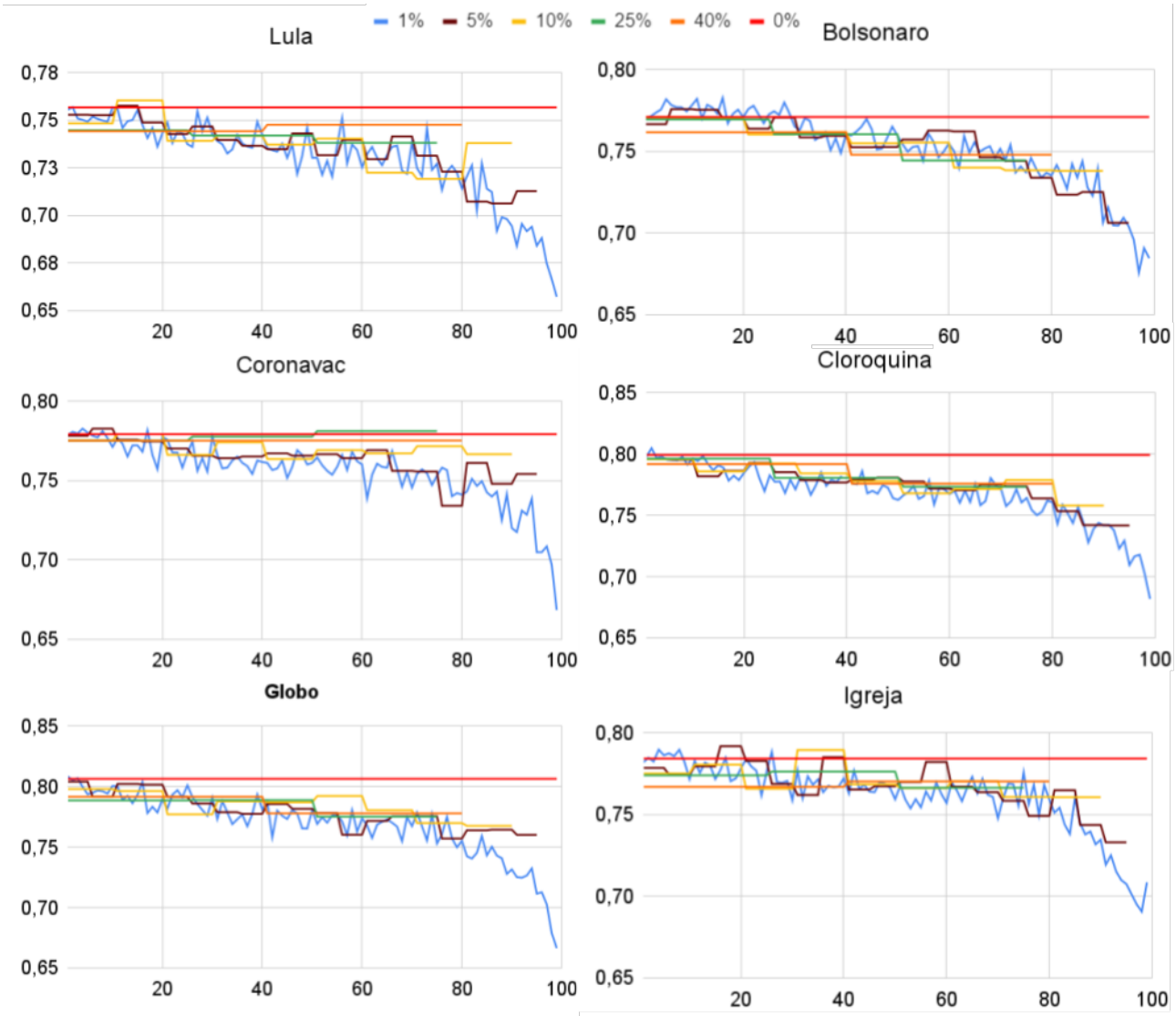


Figura 3: Expansão usando RegLog.char (horizontal: número de iterações; vertical: medida F_1.)

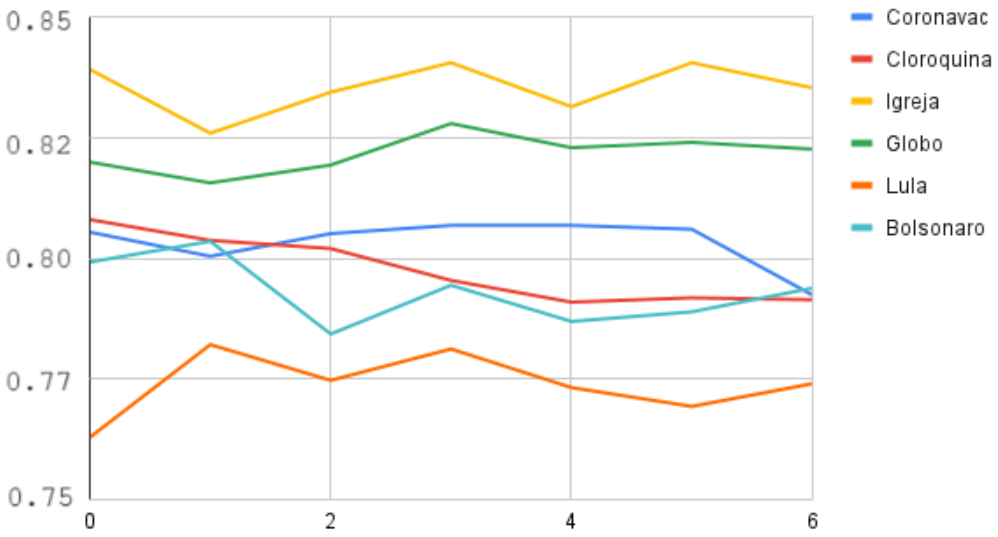


Figura 4: Expansão de dados usando BERT (horizontal: número de iterações; vertical: medida F_1).

Modelo	Alvo	F_1 Original	F_1 Expandido	Novas inst.
RegLog.word	Bolsonaro	0,75	0,78	3.520
	Lula	0,76	0,77	3.575
	Coronavac	0,77	0,78	2.864
	Cloroquina	0,80	0,80	1.760
	Globo	0,78	0,79	1.794
	Igreja	0,77	0,77	4.844
RegLog.char	Bolsonaro	0,77	0,78	5.120
	Lula	0,76	0,76	5.498
	Coronavac	0,78	0,78	3.578
	Cloroquina	0,80	0,80	704
	Globo	0,81	0,81	299
	Igreja	0,78	0,79	5.187
BERT	Bolsonaro	0,80	0,80	320
	Lula	0,76	0,78	275
	Coronavac	0,81	0,81	1.074
	Cloroquina	0,80	0,80	352
	Globo	0,81	0,83	897
	Igreja	0,84	0,85	1.730

Tabela 8: Predição de posicionamentos com base em dados originais e expandidos e quantidade de novas instâncias adicionadas pelo método de expansão.

cida aos anotadores como forma de sinalizar casos ambíguos ou que não se aplicavam às categorias a favor/contra, acarretando assim uma penalidade obrigatória para o classificador binário que não conta com esta possibilidade.

A Tabela 9 apresenta o percentual de concordância entre os anotadores e o modelo de expansão, considerando-se os dados rotulados como 'outros' como discordantes.

Alvo	% Concordância	% Outros
Bolsonaro	90,0	5,0
Lula	86,0	9,0
Coronavac	69,0	2,0
Cloroquina	97,0	0,0
Globo	91,0	9,0
Igreja	81,0	12,0

Tabela 9: Concordância entre anotador humano e modelo de expansão na avaliação manual e número de instâncias rotuladas como 'outros'.

Os resultados da Tabela 9 demonstram um bom grau de concordância para todos os tópicos, exceto 'Coronavac', em que observou-se ambiguidade significativa em alguns dos textos da amostra selecionada. Um exemplo deste tipo de ambiguidade é observado a seguir.

> '*Que boa notícia! Espero mesmo que todos sejamos vacinados ainda em janeiro! Porém, uma curiosidade: alguém sabe se outro país também vai aplicar a Coronavac na população?*'

Neste caso, observa-se que o posicionamento pode ser interpretado tanto como sendo favorável ao uso da vacina Coronavac (que já estaria disponível no mês da janeiro), quanto desfavorável (pois levanta dúvidas sobre a sua confiabilidade ao questionar se outros países também a adotariam).

Para melhor avaliar as discrepâncias entre as rotulações humanas e as produzidas pelo método apresentado, foi realizada também uma segunda anotação independente apenas destes casos problemáticos. Os resultados de concordância entre o avaliador humano original e este segundo juiz, representado pela primeira autora do presente trabalho, são apresentados na Tabela 10.

Alvo	% Concordância
Bolsonaro	50,0
Lula	85,7
Coronavac	29,0
Cloroquina	0,0
Globo	0,0
Igreja	36,8

Tabela 10: Concordância entre avaliador original e a autora da pesquisa para os casos de rotulação discrepantes.

É possível observar neste caso um grau razoável de concordância para os tópicos 'Bolsonaro' e 'Lula', o que talvez seja explicado pela maior polarização de opiniões a respeito destes dois tópicos, o que pode tornar estes posicionamentos mais facilmente detectáveis. Já a baixa concordância observada para os outros tópicos, embora influenciada pela inclusão de uma classe 'outros' não disponível ao classificador binário, sugere que a tarefa de classificação de posicionamento pode exigir grande esforço de interpretação mesmo para avaliadores humanos.

6. Considerações finais

Este artigo apresentou um método de expansão de córpus de posicionamentos em português brasileiro baseado em um *framework* existente para a tarefa correlata de análise de sentimentos (Brum & das Graças Volpe Nunes, 2018). O córpus expandido foi submetido a avaliação intrínseca e humana, com resultados indicando que a adição de instâncias rotuladas automaticamente ao córpus original não diminui a precisão da classificação, e que as instâncias adicionadas estão em sua maioria corretas.

O presente trabalho deixa uma série de oportunidades de melhoria. Dentre estas, observamos que a anotação humana tem sido utilizada apenas como ponto de partida do procedimento de expansão. No entanto, poderia ser útil, ou talvez até necessário, incluir também uma etapa de avaliação humana no ciclo de detecção-seleção, o que ajudaria a evitar a inclusão de ruído no classificador subsequente. Além disso, seria interessante comparar o presente método com alternativas recentes de anotação semiautomática de posicionamentos, fazendo uso de grandes modelos de língua, a exemplo de Zhang et al. (2023a). Iniciativas de pesquisa deste tipo são deixadas como possíveis trabalhos futuros.

Agradecimentos

Esse trabalho conta com apoio FAPESP #2021/08213-0.

Uma versão resumida deste artigo foi apresentada em Pereira & Paraboni (2024), à qual foi acrescentada uma revisão mais completa da área de detecção de posicionamentos a partir de texto (seção 2), novos experimentos de detecção de posicionamentos com uma visão mais detalhada da importância das características de aprendizado (seção 4), e uma descrição mais aprofundada dos experimentos de expansão e seus resultados, incluindo novos modelos de *baseline*, a apresentação dos resultados em forma gráfica e mais considerações sobre a tarefa de anotação humana (seção 5).

Referências

Aldayel, Abeer & Walid Magdy. 2021. Stance detection on social media: State of the art and trends. *Information Processing & Management* 58(4). 10.1016/j.ipm.2021.102597

Allaway, Emily & Kathleen R. McKeown. 2020. Zero-shot stance detection: A dataset and model using generalized topic representations. Em *Empirical Methods in Natural Language Processing (EMNLP)*, 8913–8931. 10.18653/v1/2020.emnlp-main.717

Ayyub, Kashif, Saqib Iqbal, Muhammad Wasif Nisar, Saima Gulzar Ahmad & Ehsan Ullah Munir. 2021. Stance detection using diverse feature sets based on machine learning techniques. *Journal of Intelligent & Fuzzy Systems* 40(5). 9721–9740. 10.3233/JIFS-202269

Breiman, Leo. 2001. Random forests. *Machine Learning* 45(1). 5–32. 10.1023/A:1010933404324

Brum, Henrico Bertini & Maria das Graças Volpe Nunes. 2018. Semi-supervised sentiment annotation of large corpora. Em *Conference on Computational Processing of Portuguese PROPOR*, 385–395. 10.1007/978-3-319-99722-3_39

Bugueño, Margarita & Marcelo Mendoza. 2019. Applying self-attention for stance classification. Em *Iberoamerican Congress on Pattern Recognition (CIARP)*, 51–61. 10.1007/978-3-030-33904-3_5

Carvalho, Cecília Silvestre, Vládia C Pinheiro & Lívio Freire. 2020. A deep learning model of common sense knowledge for augmenting natural language processing tasks in Portuguese language. Em *Computational Processing of the Portuguese Language*, 303–312. 10.1007/978-3-030-41505-1_29

Cavalheiro, Laís Carraro Leme, Matheus Camasmie Pavan & Ivandré Paraboni. 2023. Stance prediction from multimodal social media data. Em *Recents Advances in Natural Language Processing (RANLP)*, 242–248. 10.26615/978-954-452-092-2_027

Chen, Ninghan, Xihui Chen & Jun Pang. 2022. A multilingual dataset of COVID-19 vaccination attitudes on Twitter. *Data in Brief* 44. 108503. 10.1016/j.dib.2022.108503

Chunling, Wang, Zhang Yijia, Yu Xingyu, Liu Guantong, Chen Fei & Lin Hongfei. 2023. Adversarial network with external knowledge for zero-shot stance detection. Em *22nd Chinese National Conference on Computational Linguistics*, 824–835. 10.1007/978-981-99-6207-5_26

Conforti, Costanza, Jakob Berndt, Marco Basaldella, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd & Nigel Collier. 2021. Adversarial training for news stance detection: Leveraging signals from a multi-genre corpus. Em *EACL Hackashop on News Media Content Analysis and Automated Report Generation*, 1–7.

Conforti, Costanza, Jakob Berndt, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd & Nigel Collier. 2020. Will-They-Won't-They: A very large dataset for stance detection on Twitter. Em *58th Meeting of the Association for Computational Linguistics (ACL)*, 1715–1724. 10.18653/v1/2020.acl-main.157

da Costa, Pablo Botton, Matheus Camasmie Pavan, Wesley Ramos dos Santos, Samuel Caetano da Silva & Ivandré Paraboni. 2023. BERTabaporu: assessing a genre-specific language model for Portuguese NLP. Em *Recents Advances in Natural Language Processing (RANLP)*, 217–223. 10.26615/978-954-452-092-2_024

D'Andrea, Eleonora, Pietro Ducange, Alessio Bechini, Alessandro Renda & Francesco Marcelloni. 2019. Monitoring the public opinion about the vaccination topic from tweets analysis. *Expert Systems with Applications* 116. 209–226. 10.1016/j.eswa.2018.09.009

Darwish, Kareem, Walid Magdy & Tahar Zanouda. 2017. Improved stance prediction in a user similarity feature space. Em *Advances in Social Network Analysis and Mining (ASONAM)*, 145–148. 10.1145/3110025.3110112

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Em *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 4171–4186. 10.18653/v1/N19-1423

Fang, Wei, Moin Nadeem, Mitra Mohtarami & James Glass. 2019. Neural multi-task learning for stance prediction. Em *2nd Workshop on Fact Extraction and VERification (FEVER)*, 13–19. 10.18653/v1/D19-6603

Flores, Arthur Marçal, Matheus Camasmie Pavan & Ivandré Paraboni. 2022. User profiling and satisfaction inference in public information access services. *Journal of Intelligent Information Systems* 58(1). 67–89. 10.1007/s10844-021-00661-w

Geiss, Henri-Jacques, Flora Sakketou & Lucie Flek. 2022. OK boomer: Probing the sociodemographic divide in echo chambers. Em *10th International Workshop on Natural Language Processing for Social Media*, 83–105. 10.18653/v1/2022.socialnlp-1.8

Glandt, Kyle, Sarthak Khanal, Yingjie Li, Doina Caragea & Cornelia Caragea. 2021. Stance detection in COVID-19 Tweets. Em *59th Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 1596–1611. 10.18653/v1/2021.acl-long.127

Hanawa, Kazuaki, Akira Sasaki, Naoaki Okazaki & Kentaro Inui. 2019. Stance detection attending external knowledge from Wikipedia. *Journal of Information Processing* 27. 499–506. 10.2197/ipsjjip.27.499

Hassan, Fuad Mire & Mark Lee. 2019. Imbalanced stance detection by combining neural and external features. Em *Statistical Language and Speech Processing (SLSP)*, 273–285. 10.1007/978-3-030-31372-2_23

Jaziriyan, Mohammad Mehdi, Ahmad Akbari & Hamed Karbasi. 2021. ExaASC: A general target-based stance detection corpus in Arabic language. Em *11th International Conference on Computer Engineering and Knowledge (ICCKE)*, 424–429. 10.1109/ICCKE54056.2021.9721486

Kovačević, Aleksandar, Olivera Grljević, Zita Bošnjak & Gordana Svilengaćin. 2020. The linguistic construction of sentiment expressions in student opinionated content: A corpus-based study. *Poznan Studies in Contemporary Linguistics* 56(2). 207–249. 10.1515/psicl-2020-0006

Küçük, Dilek & Fazli Can. 2020. Stance detection: A survey. *ACM Computing Surveys* 53(1). 1–37. 10.1145/3369026

Lai, Mirko, Alessandra Teresa Cignarella, Delia Irazú Hernandez Farias, Cristina Bosco, Viviana Patti & Paolo Rosso. 2020. Multilingual stance detection in social media political debates. *Computer Speech and Language* 63. 101075. 10.1016/j.csl.2020.101075

Li, Ang, Bin Liang, Jingqian Zhao, Bowen Zhang, Min Yang & Ruifeng Xu. 2023. Stance detection on social media with background knowledge. Em *Empirical Methods in Natural Language Processing (EMNLP)*, 15703–15717. 10.18653/v1/2023.emnlp-main.972

Li, Yingjie & Cornelia Caragea. 2019. Multi-task stance detection with sentiment and stance lexicons. Em *Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 6299–6305. 10.18653/v1/D19-1657

Li, Yingjie, Tiberiu Sosea, Aditya Sawant, Ajith Jayaraman Nair, Diana Inkpen & Cornelia Caragea. 2021. P-stance: A large dataset for stance detection in political domain. Em *Findings of the Association for Computational Linguistics*, 2355–2365. 10.18653/v1/2021.findings-acl.208

Magdy, Walid, Kareem Darwish, Norah Abokhodair, Afshin Rahimi & Timothy Baldwin. 2016. #ISISisNotIslam or #DeportAllMuslims? predicting unspoken views. Em *8th Conference on Web Science*, 95–106. 10.1145/2908131.2908150

Mayfield, Elijah & Alan W Black. 2019. Stance classification, outcome prediction, and impact assessment: NLP tasks for studying group decision-making. Em *3rd Workshop on Natural Language Processing and Computational Social Science*, 65–77. 10.18653/v1/W19-2108

Mohammad, Saif, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu & Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. Em *10th International Workshop on Semantic Evaluation (SemEval-2016)*, 31–41. San Diego, California: Assoc. for Computational Linguistics. 10.18653/v1/S16-1003

Mutlu, Ece C., Toktam Oghaz, Jasser Jasser, Ege Tutunculer, Amirarsalan Rajabi, Aida Tayebi, Ozlem Ozmen & Ivan Garibay. 2020. A stance data set on polarized conversations on twitter about the efficacy of hydroxychloroquine as a treatment for COVID-19. *Data in brief* 33(106401). 10.1016/j.dib.2020.106401

Ng, Lynnette Hui Xian & Kathleen M. Carley. 2022. Is my stance the same as your stance? a cross validation study of stance detection datasets. *Information Processing & Management* 59(6). 103070. 10.1016/j.ipm.2022.103070

Pavan, Matheus Camasmie & Ivandré Paraboni. 2022. Cross-target stance classification as domain adaptation. Em *21st Mexican International Conference on Artificial Intelligence (MICAI)*, 15–25. 10.1007/978-3-031-19493-1_2

Pavan, Matheus Camasmie, Vitor Garcia dos Santos, Alex Gwo Jen Lan, Jo ao Trevisan Martins, Wesley Ramos dos Santos, Caio Deutsch, Pablo Botton da Costa, Fernando Chiu Hsieh & Ivandré Paraboni. 2023. Morality classification in natural language text. *IEEE transactions on Affective Computing* 14(1). 857–863. 10.1109/TAFFC.2020.3034050

Pavan, Matheus Camasmie, Wesley Ramos dos Santos & Ivandré Paraboni. 2020. Twitter moral stance classification using long short-term memory networks. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 636–647. 10.1007/978-3-030-61377-8_45

Pereira, Camila & Ivandré Paraboni. 2024. Semi-automatic corpus expansion: the case of stance prediction. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 515–520.

Pereira, Camila, Matheus Pavan, Sungwon Yoon, Ricelli Ramos, Pablo Costa, Laís Cavalheiro & Ivandré Paraboni. 2023. UstanceBR: a multimodal language resource for stance prediction. ArXiv [cs.CL]. 10.48550/arXiv.2312.06374

Popat, Kashyap, Subhabrata Mukherjee, Andrew Yates & Gerhard Weikum. 2019. STANCY: stance classification based on consistency cues. ArXiv [cs.CL]. 10.48550/arXiv.1910.06048

Samih, Younes & Kareem Darwish. 2021. A few topical Tweets are enough for effective user stance detection. Em *16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2637–2646. 10.18653/v1/2021.eacl-main.227

Sane, Sushmitha Reddy, Suraj Tripathi, Koushik Reddy Sane & Radhika Mamidi. 2019. Stance detection in code-mixed Hindi-English social media data using multi-task learning. Em *10th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 1–5. 10.18653/v1/W19-1301

dos Santos, Wesley Ramos & Ivandré Paraboni. 2019. Moral stance recognition and

polarity classification from Twitter and elicited text. Em *Recents Advances in Natural Language Processing (RANLP)*, 1069–1075. 10.26615/978-954-452-056-4_123

Schaefer, Robin & Manfred Stede. 2019. Improving implicit stance classification in tweets using word and sentence embeddings. Em *42nd German Conference on Artificial Intelligence*, 299–307. 10.1007/978-3-030-30179-8_26

Siddiqua, Umme Aymun, Abu Nowshed Chy & Masaki Aono. 2019. Tweet stance detection using an attention based neural ensemble model. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 1868–1873. 10.18653/v1/N19-1185

da Silva, Samuel Caetano, Thiago Castro Ferreira, Ricelli Moreira Silva Ramos & Ivandré Paraboni. 2020. Data driven and psycholinguistics motivated approaches to hate speech detection. *Computación y Systemas* 24(3). 1179–1188. 10.13053/CyS-24-3-3478

Sobhani, Parinaz, Diana Inkpen & Xiaodan Zhu. 2017. A dataset for multi-target stance detection. Em *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 551–557.

de Sousa, André & Karin Becker. 2023. SSSD: Leveraging pre-trained models and semantic search for semi-supervised stance detection. Em *14th International Conference on Recent Advances in Natural Language Processing (RANLP)*, 264–273.

Su, Zhizhong, Yaoyi Xi, Rong Cao, Huifeng Tang & Hangyu Pan. 2021. A stance detection approach based on generalized autoregressive pretrained language model in Chinese microblogs. Em *13th 13th International Conference on Machine Learning and Computing*, 232–238. 10.1145/3457682.3457717

Taulé, Mariona, Maria Antònia Martí, Francisco Manuel Rangel Pardo, Paolo Rosso, Cristina Bosco & Viviana Patti. 2017. Overview of the task on stance and gender detection in Tweets on Catalan Independence at. Em *2nd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval)*, 157–177

Tun, Yin Min & Phyu Hninn Myint. 2019. A two-phase approach for stance classification in Twitter using name entity recognition and term frequency feature. Em *18th 18th International Conference on Computer and Information Science (ICIS)*, 77–81. 10.1109/ICIS46139.2019.8940282

Wen, Haoyang & Alexander Hauptmann. 2023. Zero-shot and few-shot stance detection on varied topics via conditional generation. Em *61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 1491–1499. 10.18653/v1/2023.acl-short.127

Xu, Chang, Cecile Paris, Surya Nepal & Ross Sparks. 2019a. Recognising agreement and disagreement between stances with reason comparing networks. *arXiv preprint arXiv:1906.01392*

Xu, Zhenhui, Qiang Li, Wei Chen, Yingbao Cui, Zhen Qiu & Tengjiao Wang. 2019b. Opinion-aware knowledge embedding for stance detection. Em *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, 337–348. Springer

Zhang, Bowen, Daijun Ding, Zhichao Huang, Ang Li, Yangyang Li, Baoquan Zhang & Hu Huang. 2024. Knowledge-augmented interpretable network for zero-shot stance detection on social media. *IEEE Transactions on Computational Social Systems* 1–12. 10.1109/TCSS.2024.3388723

Zhang, Bowen, Daijun Ding & Liwen Jing. 2023a. How would stance detection techniques evolve after the launch of ChatGPT? ArXiv [cs.CL]. 10.48550/arXiv.2212.14548

Zhang, Bowen, Daijun Ding, Guangning Xu, Jinjin Guo, Zhichao Huang & Xu Huang. 2023b. Twitter stance detection via neural production systems. Em *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. 10.1109/ICASSP49357.2023.10094597

Zhang, Lei & and Bing Liu v Wang. 2018. Deep learning for sentiment analysis: A survey. *WIREs Data Mining and Knowledge Discovery* 8(4). e1253. 10.1002/widm.1253

Zhao, Xuechen, Jiaying Zou, Zhong Zhang, Feng Xie, Bin Zhou & Lei Tian. 2022. Zero-shot stance detection based on cross-domain feature enhancement by contrastive learning. ArXiv [cs.CL]. 10.48550/arXiv.2210.03380

Zhou, Shengping, Junjie Lin, Lianzhi Tan & Xin Liu. 2019. Condensed convolution neural network by attention over self-attention for stance detection in

twitter. Em *International Joint Conference on Neural Networks (IJCNN)*, 1–8. 10.1109/IJCNN.2019.8851965

Zhu, Xiaojin Jerry. 2005. Semi-supervised learning literature survey. Relatório técnico. University of Wisconsin-Madison Department of Computer Sciences.

Zhu, Zhengyuan, Kevin Meng, Josue Caraballo, Israa Jaradat, Xiao Shi, Zeyu Zhang, Farahnaz Akrami, Haojin Liao, Fatma Arslan, Damian Jimenez et al. 2021. A dashboard for mitigating the COVID-19 misinfodemic. Em *16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 99–105. 10.18653/v1/2021.eacl-demos.12

Proposta recebida em Abril 2024 e aceite para publicação em Outubro 2024.

MultiWOZ-PT: Um Conjunto de Diálogos Orientados a Tarefas em Português

MultiWOZ-PT: A Task-oriented Dialogue Dataset in Portuguese

Patrícia Ferreira
CISUC, LASI
DEI, Universidade de Coimbra

Francisco Pais
CISUC, LASI
DEI, Universidade de Coimbra

Catarina Silva
CISUC, LASI
DEI, Universidade de Coimbra

Ana Alves
CISUC, LASI
Instituto Superior de Engenharia de Coimbra

Hugo Gonçalo Oliveira
CISUC, LASI
DEI, Universidade de Coimbra

Resumo

Apesar do amplo uso da língua portuguesa, corpos de diálogos em português, disponíveis publicamente e com anotações, são escassos. Isto torna ainda mais desafiante o desenvolvimento de sistemas de diálogo eficazes que comuniquem nesta língua. Assim, apresentamos o MultiWOZ-PT, um novo conjunto de diálogos orientados a tarefas, resultado da tradução manual de uma parte do conjunto MultiWOZ para a variedade europeia do português, e da adaptação da sua base de dados. Fornecemos diretrizes abrangentes sobre o processo de criação do MultiWOZ-PT e, para demonstrar a sua utilidade prática, realizamos experiências em dois cenários orientados a tarefas: Reconhecimento de Intenções e Monitorização do Estado do Diálogo, ambos úteis para sistemas de diálogo. Os resultados obtidos ilustram a utilidade do conjunto de dados e o seu potencial para treinar e avaliar modelos de compreensão de linguagem natural e gestão de diálogo para o português. Assim sendo, o MultiWOZ-PT constitui uma contribuição significativa para o processamento computacional dessa língua, incentivando mais pesquisas e desenvolvimento de trabalho nas áreas alvo.

Palavras chave

conjunto de diálogos orientados a tarefas; tradução; multiWOZ; monitorização do estado do diálogo; reconhecimento de intenções; preenchimento de slots

Abstract

Despite the language widespread usage, publicly available and annotated Portuguese dialogue corpora are scarce. This poses a significant challenge in the development of effective dialogue systems that communicate in Portuguese. Having this in mind, we present MultiWOZ-PT, a new task-oriented dialogue dataset that results from the manual translation of dialogues in the MultiWOZ dataset to the European variety of Portuguese, as well as an adaptation of its database. We provide comprehensive guidelines and insights into the process of creating MultiWOZ-PT and, to demonstrate its practical utility, we conducted experiments in two task-oriented scenarios: Intent Recognition and Dialog State Tracking, both useful for dialogue systems. Reported results illustrate the dataset's effectiveness and its potential for training and evaluating language understanding and dialogue management models for Portuguese. Therefore, MultiWOZ-PT constitutes a significant contribution to the computational processing of this language, fostering further research and development.

Keywords

task-oriented dialogue dataset; translation; multiWOZ; dialogue state tracking; intent recognition; slot filling

1. Introdução

A tentativa de capacitar máquinas da possibilidade de se envolverem em conversas com humanos é um dos objetivos fundamentais da Inteligência Artificial e, mais especificamente, do Processamento de Linguagem Natural (PLN). Podemos dizer que *chatbots* e assistentes virtuais são a materialização deste objetivo. Especificamente, os sistemas de diálogo orientados à tarefa são projetados para ajudar os utilizadores a realizar tarefas bem definidas em diversos domínios, tais como fazer reservas em restaurantes, consultar informações meteorológicas e agendar voos, o que os torna altamente úteis para aplicações do mundo real (Zhang et al., 2020).

Corpos de dados anotados de alta qualidade

DOI: 10.21814/lm.16.2.431
This work is Licensed under a Creative Commons Attribution 4.0 License

são cruciais para o treino e avaliação de sistemas de diálogo. Desenvolver sistemas que se comunicam na língua nativa dos utilizadores é essencial para garantir acessibilidade e usabilidade, permitindo que mais pessoas interajam eficazmente com esses sistemas. Isso leva-nos a uma problema central: a escassez de corpos de dados de diálogo orientados a tarefas (CDDOTs) em idiomas que não o inglês. Quando comparados ao inglês, todos os idiomas são de recursos limitados (Wu & Dredze, 2020). Isto reflete-se na falta de recursos linguísticos para esses idiomas, e representa um desafio significativo ao limitar o desenvolvimento de sistemas orientados a dados, começando por aqueles que se baseiam em aprendizagem supervisionada.

Considerando a falta de um CDDOT disponível publicamente para o português europeu, adaptámos manualmente (parte) do Multi-Domain Wizard-of-Oz (MultiWOZ) (Budzianowski et al., 2018) para este idioma, o que resultou no MultiWOZ-PT. Este processo envolveu a adaptação da base de dados e a tradução do texto na porção de teste do MultiWOZ, que conta com 1.000 diálogos: um ficheiro com 512 e outro com 488. A base de dados original contém informações sobre serviços como hotéis, restaurantes, atrações, entre outros e foi adaptada para alinhar esses serviços, originalmente projetados para a cidade de Cambridge, Reino Unido, com o contexto de Coimbra, uma cidade em Portugal, também conhecida pela sua universidade. Para garantir consistência e qualidade dos dados traduzidos, estabelecemos algumas diretrizes, também aqui apresentadas.

O artigo demonstra ainda a utilidade prática do novo conjunto de dados. Isso é feito através da sua aplicação a duas tarefas, comuns na análise de diálogo e úteis para o desenvolvimento de sistemas de diálogo orientados a tarefas, nomeadamente: Reconhecimento de Intenções e Monitorização do Estado do Diálogo (em inglês, *Dialogue State Tracking*, DST).

Os resultados confirmam o potencial deste conjunto de dados para treinar e avaliar as tarefas-alvo em português, algo que, sem o MultiWOZ-PT, seria mais desafiante. Isso acontece porque modelos para as tarefas anteriores em português muitas vezes fazem parte de soluções proprietárias e/ou são treinados em dados proprietários, e portanto não estão disponíveis gratuitamente, nem para todos.

O restante artigo está organizado da seguinte forma. Na Secção 2, apresentamos o trabalho relacionado com foco em abordagens para criar corpos de dados anotados em português. Na Secção 3, apresentamos as diretrizes genéricas, descrevemos o conjunto de dados original MultiWOZ e a criação do MultiWOZ-PT, incluindo a sua tradução, a adaptação da base de dados e a correção de erros encontrados nas anotações originais. Na Secção 4, apresentamos e discutimos as experiências realizadas com o novo conjunto de dados, originalmente publicadas na 16ª *International Conference on Computational Processing of Portuguese* (PROPOR 2024) (Pais et al., 2024). Finalmente, na Secção 5, resumimos as principais conclusões e discutimos possíveis trabalhos futuros.

2. Trabalho relacionado

Diferentes abordagens têm sido adotadas para criar corpos de dados anotados em português. Uma passa por construí-los do zero no idioma desejado, envolvendo a recolha de documentos e sua anotação manual. Destacam-se: as coleções do HAREM (Mota & Santos, 2008; Freitas et al., 2010), onde documentos foram recolhidos de diferentes fontes e entidades mencionadas foram manualmente anotadas por especialistas na língua; ou a coleção da tarefa ASSIN (Fonseca et al., 2016), onde frases foram recolhidas e agrupadas em pares, e as suas classes de similaridade semântica e implicação foram adicionadas, com base em opiniões humanas.

A auto-anotação (em inglês, *self-labelling*) está relacionada com a anterior, mas não requer uma intervenção humana direta. É comumente adotada para corpos de dados de classificação de texto e pode depender da presença de *hashtags* específicas, por exemplo, para a classificação de ironia (da Silva, 2018); ou na fonte dos documentos, por exemplo, para a classificação de humor (Gonçalo Oliveira et al., 2020).

Uma estratégia diferente, comum para idiomas com recursos limitados, baseia-se na tradução de um conjunto de dados anotado já existente para um idioma, frequentemente inglês, para outro idioma, aproveitando as anotações existentes. Isso economiza o esforço de anotação, que leva tempo e normalmente requer especialistas ou acesso a um grande público. A tradução pode ser realizada por humanos ou com a assistência de ferramentas de tradução automática. Para o português, exemplos desta abordagem incluem a tradução de referências conhecidas para avaliar a similaridade semântica e analogia (Querido et al., 2017); ou a tradução do conjunto de dados SQuAD para resposta a perguntas (Carvalho et al., 2021).

Apesar dos exemplos anteriores, a disponibi-

lidade de corpos de dados de diálogo anotados em português é muito limitada. Por exemplo, não há um conjunto como o amplamente utilizado MultiWOZ (Budzianowski et al., 2018) ou o Schema-Guided Dialogue (SGD)(Rastogi et al., 2020), que têm conversas humanas orientadas a tarefas em inglês. Esse tipo de conjunto de dados (ou seja, CDDOT) permite aos utilizadores treinar e avaliar sistemas computacionais para uma ampla gama de tarefas no domínio do diálogo, como a geração de respostas (Gu et al., 2021), o reconhecimento de intenções (Ouyang et al., 2022; Lamanov et al., 2022), o reconhecimento de atos de diálogo (Ribeiro et al., 2019) ou a Monitorização do Estado do Diálogo (DST) (Gao et al., 2019; Siddique et al., 2021).

O MultiWOZ cobre uma diversidade de domínios e cenários, com diálogos abrangendo vários serviços, incluindo hotéis, restaurantes, atrações turísticas e transportes na cidade de Cambridge, Reino Unido. Os *slots*, que permitem armazenar informações específicas fornecidas pelos utilizadores durante a interação, estão definidos para cada serviço. O SGD referido acima também cobre diferentes domínios, cada um com esquemas e *slots* específicos. No entanto, ao contrário do MultiWOZ, cada diálogo é restrito a um único domínio.

Além do clássico MultiWOZ, este conjunto de dados tem uma versão deslexicalizada (Nekvinda & Dušek, 2021), frequentemente usada para avaliar o desempenho de sistemas de diálogo sem depender de informações específicas da base de dados. Os dados manualmente anotados capturam nuances, informações específicas e contexto detalhado, enquanto dados deslexicalizados simplificam em demasia a linguagem, resultando na perda de informação. Essa riqueza do contexto e a fidelidade ao uso da linguagem natural foi a principal razão para escolhermos dados manualmente anotados ao invés de dados deslexicalizados.

Devemos ainda referir o GlobalWOZ (Ding et al., 2021), que resulta da tradução automática do MultiWOZ para um conjunto de 20 línguas, incluindo o português. No entanto, embora a tradução automática acelere efetivamente o processo de criação, ela nem sempre será a melhor opção, como aliás mostraremos mais à frente (secção 3.3.1). Ou seja, a intervenção humana é ainda necessária para criar um conjunto de dados de qualidade, fiel a todas as nuances do diálogo e do idioma-alvo.

3. Criação do conjunto de dados

Nesta secção, descrevemos a criação do MultiWOZ-PT, um conjunto de diálogos orientados a tarefas, escritos em português. Começamos com as diretrizes seguidas para adaptar um conjunto de dados como o MultiWOZ, antes de apresentar o conjunto de dados original e o resultante, e fornecer detalhes sobre o processo de tradução e adaptação que resultou no novo conjunto de dados.

Este processo levou cerca de três meses e foi realizado por dois estudantes de Engenharia Informática (um de mestrado e um de doutoramento), autores deste artigo, falantes nativos de português europeu e com experiência em PLN, e atualmente a trabalhar em análise de diálogo. Durante este processo, reuniões com três membros seniores da equipa foram frequentes e utilizadas para rever amostras das traduções, discutir questões emergentes, e decidir como lidar com questões semelhantes no futuro.

3.1. Diretrizes gerais

Criar um novo conjunto de dados adaptado de uma fonte original é uma tarefa complexa que requer atenção meticulosa aos detalhes. Antes de iniciar o processo de tradução e adaptação, é fundamental compreender a fonte original e identificar de onde as informações serão extraídas, como bases de dados, páginas web, documentos ou outras referências relevantes.

Este trabalho teve como principal objetivo transformar um conjunto de dados em inglês, com características específicas, num conjunto de dados em português adaptado às especificidades do novo contexto linguístico, com base em fontes apropriadas e confiáveis. Para alcançar esse objetivo, traduções e adaptações precisas foram realizadas, levando em consideração as nuances da língua e da cultura.

Os tradutores humanos devem ser fluentes no idioma-alvo, pois um forte conhecimento linguístico é fundamental para manter a fidelidade e coesão no novo conjunto de dados. Além disso, entender o contexto cultural e regional-alvo é necessário para adaptar expressões, referências culturais e peculiaridades locais conforme apropriado.

Após a tradução e adaptação, deve ser realizada uma revisão abrangente das informações resultantes para garantir precisão e qualidade. Também é essencial documentar o processo de tradução e adaptação, registando decisões, recursos de referência e procedimentos específicos para ajudar a rastrear escolhas e garantir transparência no processo.

Estas diretrizes são relevantes para uma variedade de projetos de tradução e adaptação de dados, e foram aplicadas neste trabalho. Acreditamos que elas oferecem uma base sólida para garantir resultados de qualidade, independentemente da origem dos dados ou do idioma alvo.

3.2. MultiWOZ

O ponto de partida para a criação do MultiWOZ-PT foi o MultiWOZ 2.2 (Zang et al., 2020), um conhecido CDDOT. Os diálogos do MultiWOZ abrangem múltiplas interações alternadas entre dois participantes humanos: um deles assume o papel de utilizador, que tem uma tarefa a cumprir; o outro atua como um sistema, e tem como objetivo responder prontamente às solicitações do utilizador, auxiliando assim na conclusão da tarefa. É relevante notar que, no MultiWOZ, os utilizadores acreditam que estão a interagir com uma máquina (WoZ), quando, na verdade, estão a interagir com um operador humano.

A versão 2.2 do MultiWOZ foi selecionada com base na consistência e compatibilidade com modelos de dados e linguagem previamente estabelecidos, pois segue padrões de anotação para diferentes componentes dos diálogos como as intenções do utilizador e slots. Além disso, a disponibilidade de documentação e recursos facilita o desenvolvimento de pesquisas com base neste conjunto de dados. O MultiWOZ 2.2 é dividido em três partes: desenvolvimento (Desenv), treino e teste. A Tabela 1 fornece uma visão geral da distribuição do MultiWOZ 2.2 em relação à porção, número de diálogos, número de interações e média do número de interações por diálogo.

Os diálogos no MultiWOZ 2.2 abrangem um total de oito domínios: restaurante, atração, hotel, táxi, comboio, autocarro, hospital e polícia. Cada um representa uma área temática específica que molda a conversa. Outras anotações relevantes incluem intenções e *slots*. As intenções representam o objetivo subjacente por trás de uma interação do utilizador numa conversa, e os *slots* representam informações específicas que o sistema precisa de extrair das interações do utilizador para manter o contexto da conversa. O MultiWOZ original inclui 61 tipos de *slots* e tem uma estrutura organizada para as informações que podem ser solicitadas e fornecidas durante os diálogos. A Tabela 2 detalha as intenções e os *slots* no MultiWOZ. Foram utilizados os nomes originais dos *slots* e intenções, em inglês, porque não vimos necessidade para os traduzir. Assim também fica mais fácil a correspondência com as anotações do MultiWOZ original.

Além dos diálogos, inclui uma base de dados com informações sobre serviços na cidade de Cambridge, Reino Unido, relacionados aos domínios cobertos. A maioria das respostas do sistema às solicitações dos utilizadores resulta da aplicação das restrições dos utilizadores (i.e., valores dos *slots*) às informações nesta base dados.

Os *slots* servem para manter, de forma estruturada, a informação acerca do contexto. Eles podem ser categóricos ou não categóricos. Os valores dos *slots* categóricos são limitados a um conjunto de valores pré-definidos. Por exemplo, no serviço de hotel, os valores válidos para *type* e *pricerange* são respetivamente: *guesthouse* ou *hotel* e *expensive*, *cheap* ou *moderate*. Por outro lado, os *slots* não categóricos aceitam valores contínuos ou texto livre. Por exemplo, o *slot address* refere-se à morada do serviço e pode conter tanto valores numéricos quanto informações textuais.

3.3. MultiWOZ-PT

O MultiWOZ-PT resulta da tradução e adaptação do conjunto de dados original do MultiWOZ para português. Devido às limitações de tempo inerentes à metodologia que adotamos, que depende fortemente do trabalho manual, tivemos de começar por traduzir apenas parte dos diálogos originais. Nesse sentido, optamos por uma partição já existente, que poderia ter sido a de desenvolvimento ou de teste, cada uma com 1.000 diálogos. Acabou por não haver uma razão especial para termos optado pela porção de teste, mas o objetivo seria sempre, quando necessário, separar essa partição em conjuntos de treino e teste. Neste caso, a própria partição de teste é constituída por dois ficheiros que podem ser usados para o efeito[1]. Além disso, em futuros trabalhos, pretendemos expandir o MultiWOZ-PT, começando pela tradução e adaptação dos dados de desenvolvimento do MultiWOZ original.

A porção de teste do MultiWOZ 2.2 possui então 1.000 diálogos e é originalmente dividida em dois ficheiros, um com 512 diálogos e outro com 488. Na Secção 4, utilizamos esses ficheiros para treino e teste, respetivamente.

Esses 1.000 diálogos não cobrem todos os domínios, nem todos os tipos de *slot* do MultiWOZ. Eles limitam-se a cinco (restaurante, atração, hotel, táxi, comboio) dos oito domínios

[1] Admitimos que a escolha da partição de teste pode afetar a avaliação comparativa com estudos multilíngues que mantiveram o conjunto de teste original para a validação final.

2-6	#Diálogos	#Interações			#Interações
		Total	Utilizador	Sistema	/Diálogo
Treino	8.436	113.552	56.776	56.776	13,46
Desenv	1.000	14.748	7.374	7.374	14,74
Teste	1.000	14.744	7.372	7.372	14,75

Tabela 1: Distribuição do conjunto de dados MultiWOZ 2.2 nas porções de desenvolvimento (Desenv), treino e teste.

Domínio	*Slots* Categóricos	*Slots* Não Categóricos	Intenções
Restaurante	*pricerange*, *area*, *bookday*, *bookpeople*	*food*, *name*, *booktime*, *address*, *phone*, *postcode*, *ref*	*find*, *book*
Atração	*area*, *type*	*name*, *address*, *entrancefee*, *openhours*, *phone*, *postcode*	*find*
Hotel	*pricerange*, *parking*, *internet*, *stars*, *area*, *type*, *bookpeople*, *bookday*, *bookstay*,	*name*, *address*, *phone*, *postcode*, *ref*	*find*, *book*
Taxi	–	*destination*, *departure*, *arriveby*, *leaveat*, *phone*, *type*	*book*
Comboio	*destination*, *departure*, *day*, *bookpeople*	*arriveby*, *leaveat*, *trainid*, *ref*, *price*, *duration*	*find*, *book*
Autocarro	*day*	*departure*, *destination*, *leaveat*	*find*
Hospital	–	*department*, *address*, *phone*, postcode	*find*
Polícia	–	*name*, *address*, *phone*, *postcode*	*find*

Tabela 2: Mapeamento de *slots* e intenções para os domínios do MultiWOZ.

originais e a 30 tipos de *slot*. Especificamente, das 1928 interações, 399 estão relacionadas a atrações, 396 a hotéis, 445 a restaurantes, 198 a táxis e 490 a comboios. Ao traduzir, mantivemos a consistência semântica para preservar as intenções, domínios e valores dos *slots* originais. É de notar também que os números de referência originais para reservas de serviços foram preservados. No resto da secção, aprofundamos os detalhes da nossa metodologia, focando nos dois principais passos.

3.3.1. Tradução

Todas as interações na porção de teste do MultiWOZ foram traduzidas manualmente do idioma de origem, o inglês, para o português. A Tabela 3 complementa a ilustração do MultiWOZ-PT, desta vez com um diálogo completo do MultiWOZ original (em inglês) e a sua tradução para o português, no MultiWOZ-PT.

Vale ressaltar que uma tradução direta, palavra por palavra, da maioria das interações em inglês não seria viável, pois resultaria em traduções pouco naturais.

Portanto, os tradutores humanos tiveram de compreender o significado fundamental de cada interação nos diálogos originais e transmiti-lo com precisão em português, seguindo as diretrizes do mais recente Acordo Ortográfico (Silva, 2008).

Expressões idiomáticas foram adaptadas para garantir naturalidade e compreensão para os falantes de português europeu. Por exemplo, a interação "Let's grab a bite at this joint" foi traduzida para "Vamos petiscar neste local" e a interação "I'm heading downtown" foi traduzida para "Estou a caminho da baixa da cidade", refletindo melhor o contexto cultural e linguístico de Portugal. Isso garantiu que os diálogos traduzidos não só mantivessem o seu significado, mas refletissem também as nuances e peculiaridades culturais, regionais e contextuais da língua portuguesa, uma tarefa improvável de ser alcançada por meio de tradução automática. Esta abordagem centrada no ser humano é a chave para criar um conjunto de dados de alta qualidade para tarefas de PLN relacionadas a diálogos em português.

Isso é ainda mais evidente ao comparar praticamente qualquer diálogo na tradução para o português no GlobalWOZ (Ding et al., 2021) com o diálogo correspondente no MultiWOZ-PT. A Tabela 4 ilustra esse facto com algumas in-

terações retiradas de ambos os conjuntos de dados.

De forma a quantificar os problemas no GlobalWOZ, foi selecionada uma amostra aleatória de 30 diálogos deste conjunto, totalizando 402 interações, nas quais foram identificadas diversas inconsistências categorizadas em nove tipos diferentes. A Tabela 5 apresenta um resumo dessas inconsistências, indicando quantas vezes cada tipo específico foi observado nas interações de cada diálogo analisado, juntamente com um exemplo representativo de cada tipo. As inconsistências foram agrupadas em categorias como omissão de palavras e/ou frases que estão na interação original, inconsistência gramatical (inclui falta de artigos definidos, preposições incorretas, entre outros), tradução errada, inconsistência de género, ordem incorreta de palavras, falta de pontuação nas interações, contexto de Cambridge em vez de Lisboa e repetição de palavras. A média de inconsistências por interação nesta amostra é de 1,37.

Nestes 30 diálogos, verificamos que 86 interações apresentam claramente a variedade brasileira, enquanto que nós estamos a traduzir para o português europeu. Um exemplo disso é a interação "estou precisando de um trem saindo na terça-feira e chegando às 20:39". Observámos também que, em várias interações no GlobalWOZ, a interação do sistema não responde ao pedido do utilizador. Por exemplo, o utilizador pergunta "há alguma piscina a leste?" e o sistema responde "sim , existem dois Bares e discotecas .", o que indica que, em alguns casos, a base de dados de Cambridge não foi adaptada com os mesmos tipos de serviço. Este é um aspeto muito importante que foi sempre considerado na adaptação da base de dados do MultiWOZ-PT.

3.3.2. Adaptação da Base de Dados

Como mencionado anteriormente, as respostas do sistema no MultiWOZ são baseadas numa base de dados com serviços de Cambridge, Reino Unido. No entanto, ter tantos diálogos em português centrados nos serviços em Cambridge, incluindo localizações e contactos, não seria muito natural. Portanto, além de traduzirmos as interações, personalizamos os diálogos para o contexto específico da cidade portuguesa de Coimbra.

A escolha de Coimbra deve-se ao facto de ser uma cidade em Portugal, com algumas semelhanças a Cambridge, principalmente devido à sua universidade.

Não foram introduzidos tipos de *slots* além dos já existentes no conjunto de dados original. No entanto, os valores dos *slots* não categóricos foram adaptados para refletir os serviços em Coimbra. Ao adaptar restaurantes, atrações e hotéis, alteramos sistematicamente os campos essenciais, como *address*, *name*, *phone* e *postcode*, garantindo a sua consistência e relevância no novo local. As informações anteriores foram obtidas maioritariamente de plataformas conhecidas na rede, como Tripadvisor[2], principalmente para restaurantes, Booking[3], principalmente para hotéis, e também dos sites oficiais dos serviços como os sites dos próprios restaurantes, atrações ou hotéis para obter informações adicionais.

Os campos restantes foram adaptados conforme necessário para serviços específicos. Para hotéis, os campos adaptados, quando necessário, incluíram *area*, *internet*, *parking*, *pricerange*, *stars* e *type*. No caso de restaurantes, os campos adaptados, quando necessário, abrangeram *area*, *food* e *pricerange*. Para atrações, os campos adaptados incluíram *area*, *entrancefee*, *openhours*, *pricerange* e *type*. No que diz respeito aos comboios, as informações foram obtidas no sítio da CP[4], que é a plataforma da companhia ferroviária nacional de Portugal. Ele fornece informações sobre horários de comboio, rotas, preços e serviços. O nosso objetivo foi escolher comboios que tivessem como destino ou partida a cidade de Coimbra, e com durações semelhantes às da base de dados original, sempre que possível.

Para ilustrar o processo de adaptação da base de dados do MultiWOZ para o contexto de Coimbra, a Tabela 6 apresenta exemplos específicos de entradas ajustadas para dois tipos de serviços: hotel e restaurante. Esta tabela mostra uma entrada representativa para cada serviço originalmente localizado em Cambridge e como foi adaptada para a nova localização em Coimbra, mantendo a estrutura e a informação relevantes para cada categoria de serviço.

Além disso, o conjunto de dados original continha um número significativo de restaurantes indianos em Cambridge, que não são tão comuns em Coimbra. Para fazer as adaptações necessárias, seguimos duas abordagens distintas. Primeiro, adicionamos um sufixo aos nomes dos restaurantes indianos em Coimbra na nossa base de dados de restaurantes (por exemplo, *Italian Indian Palace* tornou-se *Italian Indian Palace 2* ou *Chai...*

[2] https://www.tripadvisor.pt/
[3] https://booking.com/
[4] https://www.cp.pt/

Interlocutor	**MultiWOZ 2.2**	**MultiWOZ-PT**
UTILIZADOR	I need info on a train that would be departing from Peterborough.	Preciso de informações sobre um comboio que parta da Figueira da Foz.
SISTEMA	What day and time?	A que dia e hora?
UTILIZADOR	I would like to leave on Sunday and arrive in Cambridge by 15:15.	Gostaria de partir no domingo e chegar a Coimbra pelas 15:15.
SISTEMA	I have train TR7864 leaving at 14:19 and arriving at 15:09. Would you like to book that?	Tenho o comboio 16819 com partida às 13:58 e chegada às 15:09. Gostaria de o reservar?
UTILIZADOR	That'd be perfect, I need three tickets on sunday.	Isso seria perfeito, preciso de três bilhetes para domingo.
SISTEMA	Booking was successful, the total fee is 39.59 GBP payable at the station. Reference number is: HS2FFYBI. Is there anything else I can assist you with today?	A reserva foi efetuada com sucesso, a taxa total é de 8,40 euros a pagar na estação. O número de referência é: HS2FFYBI. Posso ajudá-lo com mais alguma coisa hoje?
UTILIZADOR	I'm also looking for information on a place called the man on the moon.	Também estou à procura de informações sobre um lugar chamado Salão Brazil.
SISTEMA	The man on the moon is a concerthall in the centre. Would you like to know anything else?	O Salão Brazil é uma sala de concertos no centro. Gostaria de saber mais alguma coisa?
UTILIZADOR	Yes can I get the address to that please?	Sim, pode dar-me o endereço, por favor?
SISTEMA	Yes, the address is 2 norfolk street.	Sim, o endereço é Largo do Poço, nº3, 1º Andar.
UTILIZADOR	Can you tell me what the area of town is? Can I get the phone number too please?	Pode dizer-me qual é a zona da cidade? Pode também dar-me o número de telefone, por favor?
SISTEMA	It's located in the centre of town. The phone number is 01223474144. Do you need assistance with anything else?	Situa-se no centro da cidade. O número de telefone é 239837078. Precisa de ajuda com mais alguma coisa?
UTILIZADOR	That will be it for today! Thank you so much!	É tudo por hoje! Muito obrigado!
SISTEMA	You're very welcome! Have a great day!	Não tem de quê! Tenha um ótimo dia!

Tabela 3: ID do diálogo original PMUL1241 do MultiWOZ e a sua tradução no MultiWOZ-PT.

Conjunto de dados	**Interação**
MultiWOZ	Please find a moderate price range Turkish restaurant.
GlobalWOZ	encontre um restaurante de faixa de preço preço moderado Europeia .
MultiWOZ-PT	Por favor, procure um restaurante turco de preço moderado.
MultiWOZ	I want to book a table for 3 people at 13:00 on wednesday will you do that ?
GlobalWOZ	eu quero reservar uma mesa para pessoas Sete em 19:00 em quarta-feira .
MultiWOZ-PT	Quero reservar uma mesa para 3 pessoas às 13:00 de quarta-feira, podem fazê-lo?
MultiWOZ	Would you help me book a train today?
GlobalWOZ	você me ajudaria a reservar um trem hoje ?
MultiWOZ-PT	Poderia ajudar-me a reservar um comboio hoje?
MultiWOZ	I would be happy to. Where are you traveling?
GlobalWOZ	Eu ficaria feliz em .
MultiWOZ-PT	Teria todo o gosto. Para onde viaja?

Tabela 4: Interações no MultiWOZ original (diálogos PMUL3731 e PMUL1623), GlobalWOZ e MultiWOZ-PT.

Inconsistência	**Contagem**	**Contagem (%)**	**Exemplo da interação com inconsistência e da interação original do MultiWOZ**
Omissão de parte da frase	216	46.65	train PL020 atenderia aos seus critérios. Train TR4803 would meet your criteria. Can I book something for you?
Inconsistência gramatical	132	28.51	posso fazer uma reserva para nove no 17:30 nesse quarta-feira? Can I book a reservation for 7 at 13:00 this Friday?
Tradução errada	37	7.99	por favor, encontre um lugar para ir na leste e deve ser uma colagem. please find me a place to go in the centre and it should be a collage.
Inconsistência de género	33	7.13	claro, há dois na Oeste. Sure there are two in the centre of town. I prefer Anatólia located at 30 Bridge Street City Centre.
Ordem incorreta	20	4.32	sim, para pessoas Sete. Yes for 6 people. 2 nights starting from saturday.
Falta de pontuação	9	1.94	Preciso reservar uma mesa para o mesmo grupo de pessoas em 20:00 no terça em um restaurante Europeia. I need to book a table for the same group of people at 12:30 on the same day at a mexican restaurant.
Contexto de Cambridge	8	1.73	estou procurando um lugar em Cambridge. I am looking for a place in Cambridge. It doesn't need to include internet and should be a hotel
Repetição de palavras	8	1.73	estarei entrando na central de central e quero chegar por volta de central. I'll be coming into Cambridge from Bishops Stortford.

Tabela 5: Exemplos e contagem de inconsistências encontradas numa amostra de 30 diálogos do GlobalWOZ e a respetiva interação no MultiWOZ original.

Serviço	**MultiWOZ 2.2**	**MultiWOZ-PT**
Hotel	name: ashley hotel address: 74 chesterton road area: north internet: yes parking: yes phone: 01223350059 postcode: cb41er pricerange: moderate stars: 2 type: hotel	name: Hotel Mondego address: Largo das Ameias 3-4 area: north internet: yes parking: no phone: 239496239 postcode: 3000-024 pricerange: moderate stars: 2 type: hotel
Restaurante	name: ask restaurant address: 12 Bridge Street City Centre area: centre food: italian phone: 01223364917 postcode: cb21uf pricerange: cheap type: restaurant	name: La Divina Pizza Bar address: Rua Carlos Seixas 267 area: centre food: italian phone: 239093914 postcode: 3030-177 pricerange: cheap type: restaurant

Tabela 6: Exemplos de serviços da base de dados do MultiWOZ para Cambridge e as suas adaptações no MultiWOZ-PT para Coimbra.

tornou-se *Chai2...*). Uma segunda opção envolveu a substituição de referências a certos restaurantes indianos por estabelecimentos de outro tipo mais prevalentes em Coimbra (por exemplo, o restaurante indiano *pipasha restaurant* foi substituído por um restaurante português "Casas do Bragal").

3.3.3. Correção dos procedimentos

Algumas palavras com erros ortográficos em inglês foram identificadas no conjunto de dados original. Por exemplo, a palavra *avaliable* na interação *There are several guesthouses avaliable. (...)* (diálogo MUL2155) ou a palavra *adress* na interação *get me the adress please* (diálogo PMUL3282). Apesar de ser importante que os sistemas de diálogo sejam tolerantes a este tipo de erros, eles não foram replicados na tradução para o português e as palavras foram corretamente traduzidas.

Foi ainda identificado um erro sistemático no conjunto de dados original do MultiWOZ: um total de 975 interações de utilizador foram rotuladas com múltiplas intenções, embora, na maioria dos casos, isso não devesse ocorrer. Por exemplo, a interação *I need a train from Norwich to Cambridge on Friday, please. I need to get there at about 10:30.* (diálogo PMUL4034) foi rotulada com as intenções *find-train* e *find-hotel*, embora a última fosse claramente incorreta. Essas inconsistências foram prontamente corrigidas, o que explica a disparidade no número de intenções entre o conjunto de dados original e o conjunto de dados em português, na Tabela 7.

4. Experimentação

Nesta secção, demonstramos a utilização do MultiWOZ-PT em duas tarefas relacionadas com análise de diálogo, nomeadamente: Reconhecimento de Intenções e Monitorização de Estado de Diálogo (DST, em inglês). Embora reconheçamos que existem diversas outras tarefas importantes no domínio do diálogo, escolhemos estas pois o MultiWOZ oferece suporte de anotação e é frequentemente utilizado para avaliá-las (Lamanov et al., 2022; Tavares et al., 2023; Zhang et al., 2019). Com o novo conjunto de dados, ficam disponíveis e adaptadas para o contexto da língua portuguesa, tornando-se mais acessíveis à comunidade.

4.1. Reconhecimento de Intenções

O Reconhecimento de Intenções é uma tarefa comum em sistemas de diálogo que tem como objetivo identificar aquilo que o utilizador expressa querer realizar. Assim, desempenha um papel crucial em aprimorar a experiência do utilizador.

As frases do MultiWOZ-PT estão anotadas com oito intenções diferentes, apresentadas na Tabela 7. Portanto, esta experiência envolveu o treino de um modelo para classificar automaticamente essas intenções. Para esse fim, utilizamos como ponto de partida dois modelos da família BERT disponíveis, nomeadamente: BERTimbau-base (Souza et al., 2020) e Albertina-PTPT (Rodrigues et al., 2023). Eles foram utilizados através da biblioteca *transformers*, em Python, e do repositório HuggingFace[5][6]. Ambos os modelos são dedicados ao português. No entanto, o primeiro, que está disponível há mais tempo, foi pré-treinado a partir de textos em português do Brasil. O segundo, Albertina-PTPT, é baseado no DeBERTa (He et al., 2020) e foi pré-treinado a partir de textos em português europeu, a mesma variedade do MultiWOZ-PT.

Para realizar o Reconhecimento de Intenções, os modelos anteriores foram afinados (em inglês, *fine-tuned*) nas anotações de intenção do MultiWOZ-PT. Para uma divisão mais natural, nesta fase usamos o primeiro ficheiro, que contém 512 diálogos. Os seguintes hiperparâmetros foram usados para a afinação de ambos os modelos: *batch size* 32, *learning rate* de $1e^{-5}$ e duração do treino de 5 épocas. O desempenho foi medido nos 488 diálogos restantes.

Além dos modelos baseados em BERT, também implementámos um modelo de máquina de vetores de suporte (SVM) linear com vetores Frequência do Termo-Inverso da Frequência nos Documentos (Tf-idf). Utilizou-se o SVM com kernel linear, mantendo os parâmetros padrão. Na implementação do Tf-idf, consideraram-se todos os unigramas que aparecem em pelo menos dois documentos (min_df = 2), com os restantes parâmetros configurados como padrão. Ambos estão disponíveis na biblioteca *scikit-learn*[7][8].

[5]BERTimbau disponível em `huggingface.co/neuralmind/bert-base-portuguese-cased`

[6]Albertina disponível em `huggingface.co/PORTULAN/albertina-ptpt`

[7]SVM disponível em `https://scikit-learn.org/stable/modules/svm.html`

[8]Tf-idf disponível em `https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html`

	MultiWOZ		MultiWOZ-PT	
Intents	#	%	#	%
find_attraction	1.232	17,10	1.065	17,10
find_hotel	1.235	17,18	1.093	17,57
book_hotel	483	6,71	432	6,93
find_restaurant	1.183	16,42	1.035	16,62
book_restaurant	580	8,06	440	7,06
find_taxi	495	6,86	416	6,68
find_train	1.562	21,69	1.379	22,17
book_train	431	5,98	366	5,87
Total	7.201	100	6.226	100

Tabela 7: Intenções no conjunto de teste para o MultiWOZ e para o MultiWOZ-PT.

A Tabela 8 reporta a precisão (P), abrangência (R) e medida F1 (F1) dos modelos, quando aplicados a esses diálogos, em média e distribuídos por intenção.

Os modelos demonstram consistentemente um desempenho forte na tarefa de Reconhecimento de Intenções, alcançando a precisão, abrangência e medida F1 superiores ou igual a 0,77. O modelo SVM linear utilizando TF-IDF tem um desempenho ligeiramente mais baixo que os restantes que têm desempenhos semelhantes. O principal objetivo desta experiência é demonstrar o MultiWOZ-PT no Reconhecimento de Intenções orientadas em diálogos orientados a tarefas, onde pode ser usado para treinar e avaliar modelos. Como se pode verificar na Tabela 7, o conjunto de dados não é equilibrado em termos de intenções, o que pode ter impacto no desempenho. Por exemplo, os modelos Albertina-PTPT e BERTimbau têm um desempenho melhor para intenções como *find_train* (F1$\approx$ 0,90), seguido por *find_attraction* e *find_hotel*, que são as três intenções mais representadas. Por outro lado, o desempenho é pior para *book_hotel* e *book_train*, as menos representadas.

4.2. DST

A maioria dos sistemas de diálogo orientados a tarefas adota uma arquitetura conhecida como Estado do Diálogo (em inglês, *Dialog State*), que compreende a Monitorização do Estado do Diálogo (DST), uma Política de Diálogo e Atos de Diálogo. O DST (Williams et al., 2016) acompanha o estado atual de uma conversa preenchendo *slots* específicos. Por exemplo, supondo que pretendemos preencher os slots *food* e *area* e que o utilizador refere a frase "Estou à procura de um restaurante europeu na zona oeste de Coimbra", podemos preencher o slot *food* com o valor "europeu" e o slot *area* com o valor "oeste". Isto realiza-se para todos os turnos do diálogo.

O objetivo desta segunda experiência é realizar DST no MultiWOZ-PT, o que envolve extrair *slots* e os seus valores a partir das interações dos utilizadores. A experiência é descrita em várias subsecções, onde também explicamos diferentes etapas da abordagem adotada.

4.2.1. Resposta a Perguntas para Preenchimento de Slots

Abordamos o preenchimento de *slots* como uma tarefa de Resposta Automática a Perguntas (QA), que pode aproveitar modelos disponíveis baseados na arquitetura *transformer*, como o BERT (Devlin et al., 2019) e o T5 (Raffel et al., 2020), afinados em corpos de dados de QA. Dada uma pergunta em linguagem natural e um contexto, estes modelos extraem a sequência no contexto que melhor responde à pergunta[9]. O preenchimento de cada *slot* baseia-se depois na realização uma pergunta em linguagem natural, utilizando a interação como contexto.

Para QA utilizamos uma versão do BERTimbau afinada numa tradução do conjunto de dados SQuAD (Rajpurkar et al., 2016) para o português, um modelo disponível no portal Huggingface[10]. O SQuAD contém mais de 100.000 pares de perguntas e respostas em mais de 500 artigos da Wikipédia. Como não é específico de um domínio, ele adapta-se bem a diferentes contextos, principalmente quando as perguntas feitas são simples e diretas.

[9]Mais propriamente, o BERT identifica o início e o fim da sequência que responde à pergunta, enquanto que o T5 gera essa sequência. Como, no SQuAD, as respostas são sempre sequências que ocorrem no contexto, é expectável que a sequência gerada pelo T5 também esteja no contexto, ainda que isso não seja garantido.

[10]https://huggingface.co/pierreguillou/bert-large-cased-squad-v1.1-portuguese

Intenção	Albertina-PTPT			BERTimbau			SVM com Tf-idf		
	P	R	F1	P	R	F1	P	R	F1
find_attraction	0,76	0,90	0,83	0,81	0,90	0,85	0,71	0,89	0,79
find_hotel	0,80	0,83	0,81	0,81	0,84	0,82	0,85	0,77	0,80
book_hotel	0,72	0,76	0,74	0,78	0,75	0,76	0,89	0,68	0,77
find_restaurant	0,84	0,77	0,80	0,87	0,75	0,81	0,75	0,77	0,76
book_restaurant	0,78	0,88	0,83	0,72	0,84	0,78	0,84	0,65	0,73
find_taxi	0,95	0,74	0,83	0,80	0,82	0,81	0,96	0,77	0,85
find_train	0,92	0,89	0,90	0,91	0,88	0,89	0,84	0,92	0,88
book_train	0,83	0,63	0,72	0,79	0,75	0,77	0,71	0,71	0,71
Média macro	0,82	0,80	0,81	0,81	0,82	0,81	0,82	0,77	0,79
Média ponderada	0,83	0,83	0,83	0,83	0,83	0,83	0,81	0,80	0,80

Tabela 8: Desempenho do Reconhecimento de Intenções no MultiWOZ-PT.

Além do modelo, é necessário ter uma pergunta em linguagem natural para cada um dos 30 tipos de *slot* no MultiWOZ-PT. As perguntas foram criadas manualmente, mas isso só tem de ser feito uma vez para cada conjunto de dados/*slots*. As perguntas foram o resultado de testes preliminares com o intuito de seguir um formato semelhante ao do SQuAD, garantindo que fossem diretas e mencionando sempre o nome do slot. Apesar de envolver trabalho manual, tem apenas de ser feito uma vez, e eventualmente revisto quando os tipos de *slot* mudam, por exemplo, para novos diálogos ou para um novo conjunto de dados. A Tabela 9 mostra as perguntas usadas para os *slots* do serviço de hotel.

A Figura 1 ilustra a abordagem adotada que representa uma contribuição deste trabalho. Dada uma pergunta ("Qual o nome do Hotel?") e uma interação ("Olá, gostaria de ficar no Hotel Astória.") como entrada, o modelo extrai a resposta desejada ("Hotel Astória"), que é então usada para preencher um *slot*, neste caso, *hotel_name*.

À sequência que responde à pergunta, os modelos de QA juntam um valor entre 0 e 1, num campo "score", associado à confiança na resposta dada. Tiramos partido desse valor para ignorar respostas cuja confiança é mais baixa. Na maior parte dos casos, estas correspondem a perguntas para as quais a interação, dada como contexto, não tem uma resposta. Na Figura 1, está um caso em que se faz uma pergunta para a qual a resposta existe e, por essa razão, a confiança é alta (0,992). Por outro lado, se fosse feita uma pergunta para a qual não há resposta, por exemplo, se a pergunta "Em que dia é a reserva?" for feita ao contexto da figura, é normal obter uma resposta com baixa confiança, tal como gostaria de ficar no "Hotel Astória", com "score" 0,069.

Embora não seja a abordagem mais comum, uma vantagem é que esta se baseia em modelos de

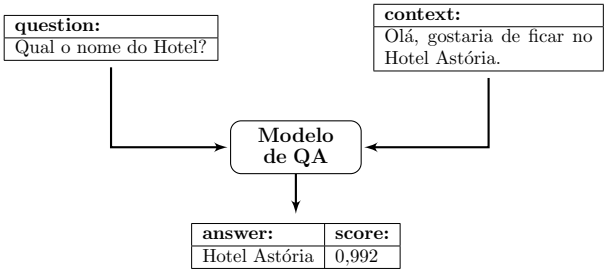


Figura 1: Exemplo do modelo de QA em funcionamento.

QA genéricos disponíveis e não requer supervisão em dados de DST. O trabalho manual necessário é mínimo, o que permite a aplicação da abordagem a cenários onde ou não há dados anotados disponíveis, ou não são suficientes para o treino. Ressalvamos que o MultiWOZ-PT possui 30 tipos diferentes de *slots* e treinar um modelo para preenchê-los exigiria mais dados do que para o Reconhecimento de Intenções.

4.2.2. Pós-processamento para Slots Categóricos

Entre as perguntas respondidas de forma imprecisa, algumas estão suficientemente próximas dos valores válidos dos *slots*. Isto acontece, por exemplo, porque os utilizadores podem usar variações como sinónimos; ou porque o modelo de QA pode extrair informações adicionais. Para mapear as respostas anteriores em valores válidos dos *slots*, é aplicado um pós-processamento aos valores dados para os *slots* categóricos.

Dois métodos diferentes foram testados para associar as respostas aos valores mais próximos para esses *slots*, nomeadamente:

- Distância de Levenshtein, baseada no número mínimo de edições de caracteres necessárias para transformar a resposta num valor válido;

Tipo de *Slot*	Pergunta
hotel-area	*Em que área está localizado o estabelecimento?*
hotel-bookday	*Em que dia é a reserva?*
hotel-bookpeople	*Quantas pessoas são?*
hotel-bookstay	*Quantos dias vai ficar?*
hotel-internet	*Tem internet grátis?*
hotel-name	*Qual é o nome do estabelecimento?*
hotel-parking	*Tem estacionamento gratuito?*
hotel-pricerange	*Qual é o preço médio do estabelecimento?*
hotel-stars	*Quantas estrelas tem?*
hotel-type	*Qual é o tipo de estabelecimento?*

Tabela 9: Perguntas para o Serviço de Hotel.

- Similaridade Textual Semântica (STS), baseada na similaridade do cosseno entre as representações vetoriais (em inglês, *embeddings*) da resposta e cada valor válido. As representações foram obtidas a partir de um *transformer* disponível na HuggingFace, baseado na afinação do BERTimbau nos pares das coleções ASSIN[11].

A Tabela 10 ilustra o pós-processamento com exemplos onde as respostas foram mapeadas com sucesso para valores válidos de *slot*. Sem o pós-processamento, os resultados obtidos seriam considerados incorretos.

Método	*Slot* Original	*Slot* preenchido
Levenshtein	arquitetura	arquitetónico
	barco	barcos
	caro	cara
STS	meio dia	depois do meio dia
	sexta-feira	sexta-feira às 16:00
	centro	centro da cidade

Tabela 10: Exemplos de comparações corretamente avaliadas utilizando os métodos Levenshtein e STS como etapa de pós-processamento.

4.2.3. Limitação do Número de Respostas

Independentemente da pergunta, havendo um contexto, o modelo de QA extrairá sempre uma resposta. Portanto, fazer todas as 30 perguntas para cada interação do utilizador é propício a produzir ruído. Uma primeira verificação na qualidade da resposta baseia-se na pontuação que o modelo atribui à resposta (*score*), um número no intervalo de 0 a 1 que reflete a confiança. Ao definir um limite na pontuação, podemos limitar o número de respostas a serem consideradas. Um limite alto deve manter as respostas boas e descartar as de qualidade inferior, possivelmente incorretas. Um limite baixo pode ser muito restritivo e resultar em nenhuma resposta considerada. Para determinar os limites que maximizam o desempenho, testamos valores na gama de 0,49 a 0,79. Tal como na experiência anterior, isso foi feito nos primeiros 512 diálogos do MultiWOZ-PT.

Importa referir que o Reconhecimento de Intenções pode ser útil para restringir o conjunto de perguntas àquelas relacionadas com a intenção alvo. Portanto, para identificar as intenções do utilizador, usamos um dos classificadores descritos na Secção 4.1. Como o desempenho foi semelhante, escolhemos o classificador baseado no BERTimbau, que é também o modelo com menos parâmetros.

4.2.4. Avaliação

A avaliação do DST recorre geralmente a duas métricas, nomeadamente:

- A *Joint Goal Accuracy* (JGA) mede a compreensão geral do sistema em relação à intenção do utilizador. É uma métrica "tudo ou nada" que procura avaliar se o sistema compreende efetivamente todos os objetivos do utilizador.
- A *Slot* F1 avalia a capacidade do sistema de identificar com precisão os valores dos *slots* associados a cada interação

A abordagem baseada em QA foi testada na segunda parte do MultiWOZ-PT, compreendendo 488 diálogos. A Tabela 11 relata o seu desempenho ao aplicar cada um dos métodos de pós-processamento ou nenhum.

O DST é desafiador e os valores de referência no MultiWOZ são inferiores a 0,60 (Heck et al., 2020; Lee et al., 2021). Como a abordagem adotada não foi supervisionada nesta tarefa, consideramos os resultados obtidos como interes-

[11] huggingface.co/rufimelo/bert-large-portuguese-cased-sts

santes e como uma base que pode ser melhorada no futuro. Concluímos também que o pós-processamento é benéfico e que ambos os métodos têm uma JGA comparável, ainda que a distância de Levenshtein leve a pontuações de *Slot* F1 superiores.

Diferentes serviços apresentam desafios diferentes e ambas as métricas variam entre eles. O melhor desempenho é alcançado para os comboios. O pior depende da métrica e é para restaurante (JGA) e táxi (*Slot* F1).

5. Conclusões e Trabalho Futuro

Apresentamos o MultiWOZ-PT, um conjunto de dados de diálogo orientados a tarefas criado a partir do conjunto MultiWOZ, amplamente utilizado, que adaptámos para o contexto da língua e cultura portuguesas. Este novo conjunto de dados é uma resposta à ausência de um CDDOT de qualidade e publicamente disponível em português. Os diálogos e a base de dados do MultiWOZ-PT estão disponíveis para todos os interessados no repositório GIT[12] e no repositório OSF[13], onde também está incluído o código produzido para a realização das experiências aqui descritas.

O processo de adaptação envolveu: (i) a tradução e contextualização da parte de teste do MultiWOZ; e (ii) a adaptação da base de dados original, onde que serviços em Cambridge, Reino Unido, foram substituídos por serviços semelhantes em Coimbra, Portugal.

Além de apresentarmos o MultiWOZ-PT, realizámos duas experiências que ilustram possíveis aplicações do conjunto de dados e confirmam que ele abre portas a tarefas relacionadas com o processamento de diálogos em português. Começámos com o Reconhecimento de Intenções, um passo comum na maioria dos sistemas de diálogo, útil para diferenciar o tratamento de diferentes pedidos e consequentes respostas. Em seguida, abordámos o DST, útil para representar o contexto de um diálogo orientado a tarefas, ao procurar respostas plausíveis, considerando mais do que apenas a última interação. Ambas as experiências foram realizadas com modelos reconhecidos e confirmaram a coerência do novo conjunto de dados.

Este trabalho aborda uma lacuna crítica no processamento computacional da língua portuguesa, atuando como um catalisador para futuras pesquisas e avanços. Como um conjunto de dados publicamente disponível, o MultiWOZ-PT permite o treino e/ou a avaliação de novos modelos para as tarefas anteriores ou outras relacionadas com o processamento de diálogo na língua portuguesa. Assim, esperamos estar a contribuir para o avanço do estado da arte. Com um melhor processamento da língua portuguesa, estamos a beneficiar não apenas os falantes dessa língua, mas também a fortalecer o desenvolvimento de tecnologias mais acessíveis e eficazes.

Tendo em vista uma referência mais abrangente, o trabalho futuro inclui o aumento do tamanho do conjunto de dados. Isto exigirá a tradução de mais diálogos, ao mesmo tempo que serão feitas adaptações necessárias na base de dados. Em vez de seguir o mesmo processo, podemos explorar abordagens automáticas de adaptação, como a do GlobalWOZ (Ding et al., 2021). No entanto, ainda é nossa intenção tratar manualmente os diálogos e a base de dados resultantes, e aproveitar essa abordagem apenas para acelerar o processo. Na verdade, com os problemas observados na tradução para o português do GlobalWOZ, isso é altamente recomendável.

Como mostramos, o MultiWOZ-PT pode suportar diferentes tarefas relacionadas à análise de diálogos em português. Os resultados apresentados aqui podem ser vistos como linhas de base e têm espaço para melhorias, algo que também planeamos abordar no futuro. Por exemplo, o Reconhecimento de Intenções pode beneficiar da consideração das intenções anteriores na classificação (por exemplo, adicionando uma camada CRF aos modelos atuais). Quanto ao DST, o próximo passo seria testar abordagens supervisionadas no MultiWOZ-PT, começando com aquelas já aplicadas ao MultiWOZ (Heck et al., 2020; Lee et al., 2021).

Algumas das abordagens anteriores são baseadas na afinação de um modelo de linguagem para as respetivas tarefas. Aqui, uma experiência interessante seria analisar o impacto do treino de modelos numa quantidade maior de diálogos traduzidos automaticamente por oposição a uma quantidade menor de diálogos traduzidos manualmente. Para garantir uma avaliação justa, os modelos seriam sempre comparados em diálogos traduzidos manualmente, como os que já temos.

Por fim, também estamos interessados em analisar fluxos comuns em diálogos em português de diferentes tipos, com múltiplos propósitos. Aqui, mais do que fazer essa análise no MultiWOZ-PT, o novo conjunto de dados pode apoiar alguns de nossos estudos. Mais precisamente, modelos treinados neste conjunto de dados podem ser usados para enriquecer diálogos

[12] https://github.com/NLP-CISUC/Dialog-State-Tracking-PT

[13] https://osf.io/vxq6p/

	JGA			*Slot* F1		
Serviços	**Nenhum**	**Lev**	**STS**	**Nenhum**	**Lev**	**STS**
Atração	0,25	0,36	**0,37**	0,46	**0,53**	0,52
Hotel	0,27	**0,29**	0,27	0,50	**0,52**	0,45
Restaurante	0,19	**0,22**	0,20	0,52	**0,54**	0,50
Taxi	0,32	0,35	**0,39**	0,47	**0,51**	0,50
Comboio	0,30	**0,51**	0,32	0,65	**0,72**	0,56
Média macro	0,27	0,35	0,31	0,52	0,56	0,51
Média ponderada	0,26	0,32	0,29	0,54	0,58	0,50

Tabela 11: Desempenho do DST no MultiWOZ-PT, com reconhecimento de intenções integrado e diferentes métodos de pós-processamento.

em bruto (por exemplo, com intenções ou *slots*), facilitando assim a identificação de tendências e problemas de comunicação (por exemplo, numa pós-análise).

Agradecimentos

Este trabalho foi parcialmente apoiado pelo Plano de Recuperação e Resiliência (PRR) Português através do projeto C645008882-00000055, Centro para a IA Responsável; pela FCT – Fundação para a Ciência e a Tecnologia, I.P., no âmbito do projeto CISUC – UID/CEC/00326/2020 e pelo Fundo Social Europeu, através do Programa Operacional Regional Centro 2020.

Referências

Budzianowski, Paweł, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan & Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5016–5026. 10.18653/v1/D18-1547

Carvalho, Nuno Ramos, Alberto Simões & José João Almeida. 2021. Bootstrapping a dataset and model for question-answering in Portuguese. Em *10th Symposium on Languages, Applications and Technologies (SLATE)*, 18:1–18:5. 10.4230/OASIcs.SLATE.2021.18

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 4171–4186. 10.18653/v1/N19-1423

Ding, Bosheng, Junjie Hu, Lidong Bing, Sharifah Mahani Aljunied, Shafiq Joty, Luo Si & Chunyan Miao. 2021. GlobalWoZ: Globalizing multiwoz to develop multilingual task-oriented dialogue systems. ArXiv [cs.CL]. 10.48550/arXiv.2110.07679

Fonseca, E, L Santos, Marcelo Criscuolo & S Aluísio. 2016. ASSIN: Avaliação de similaridade semântica e inferencia textual. Em *Computational Processing of the Portuguese Language (PROPOR)*, 13–15.

Freitas, Cláudia, Paula Carvalho, Hugo Gonçalo Oliveira, Cristina Mota & Diana Santos. 2010. Second HAREM: advancing the state of the art of named entity recognition in Portuguese. Em *International Conference on Language Resources and Evaluation (LREC)*,

Gao, Shuyang, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung & Dilek Hakkani-Tur. 2019. Dialog state tracking: A neural reading comprehension approach. ArXiv [cs.CL]. 10.48550/arXiv.1908.01946

Gonçalo Oliveira, Hugo, André Clemêncio & Ana Alves. 2020. Corpora and baselines for humour recognition in Portuguese. Em *12th Language Resources and Evaluation Conference (LREC)*, 1278–1285.

Gu, Xiaodong, Kang Min Yoo & Jung-Woo Ha. 2021. DialogBERT: Discourse-aware response generation via learning to recover and rank utterances. Em *AAAI Conference on Artificial Intelligence*, 12911–12919. 10.1609/aaai.v35i14.17527

He, Pengcheng, Xiaodong Liu, Jianfeng Gao & Weizhu Chen. 2020. DeBERTa: Decoding-enhanced bert with disentangled attention. ArXiv [cs.CL]. 10.48550/arXiv.2006.03654

Heck, Michael, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin,

Marco Moresi & Milica Gasic. 2020. TripPy: A triple copy strategy for value independent neural dialog state tracking. Em *21st Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 35–44. 10.18653/v1/2020.sigdial-1.4

Lamanov, Dmitry, Pavel Burnyshev, Katya Artemova, Valentin Malykh, Andrey Bout & Irina Piontkovskaya. 2022. Template-based approach to zero-shot intent recognition. Em *15th International Conference on Natural Language Generation*, 15–28. 10.18653/v1/2022.inlg-main.2

Lee, Chia-Hsuan, Hao Cheng & Mari Ostendorf. 2021. Dialogue state tracking with a language model using schema-driven prompting. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4937–4949. 10.18653/v1/2021.emnlp-main.404

Mota, Cristina & Diana Santos (eds.). 2008. *Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O segundo HAREM*. Linguateca.

Nekvinda, Tomáš & Ondřej Dušek. 2021. Shades of BLEU, flavours of success: The case of MultiWOZ. Em *1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, 34–46. 10.18653/v1/2021.gem-1.4

Ouyang, Yawen, Zhen Wu, Xinyu Dai, Shujian Huang & Jiajun Chen. 2022. Towards multi-label unknown intent detection. Em *29th International Conference on Computational Linguistics (COLING)*, 626–635.

Pais, Francisco, Patrícia Ferreira, Catarina Silva, Ana Alves & Hugo Gonçalo Oliveira. 2024. Question answering for dialogue state tracking in Portuguese. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 461–471.

Querido, Andreia, Rita Carvalho, João Rodrigues, Marcos Garcia, João Silva, Catarina Correia, Nuno Rendeiro, Rita Valadas Pereira, Marisa Campos & António Branco. 2017. LX-LR4DistSemEval: A collection of language resources for the evaluation of distributional semantic models of Portuguese. *Revista da Associação Portuguesa de Linguística* 3. 265–283. 10.26334/2183-9077/rapln3ano2017a15

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li & Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* 21(1). 5485–5551.

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev & Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. ArXiv [cs.CL]. 10.48550/arXiv.1606.05250

Rastogi, Abhinav, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta & Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The Schema-Guided Dialogue Dataset. Em *AAAI Conference on Artificial Intelligence*, vol. 34 05, 8689–8696. 10.1609/aaai.v34i05.6394

Ribeiro, Eugénio, Ricardo Ribeiro & David Martins de Matos. 2019. Reconhecimento de actos de diálogo hierárquicos e multi-etiqueta em dados em Espanhol. *Linguamática* 11(1). 17–40. 10.21814/lm.11.1.278

Rodrigues, João, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso & Tomás Osório. 2023. Advancing neural encoding of Portuguese with transformer Albertina PT-*. ArXiv [cs.CL]. 10.48550/arXiv.2305.06721

Siddique, AB, Fuad Jamour & Vagelis Hristidis. 2021. Linguistically-enriched and context-awarezero-shot slot filling. Em *The Web Conference (WWW)*, 3279–3290. 10.1145/3442381.3449870

da Silva, Fábio Ricardo Araújo. 2018. *Detecção de ironia e sarcasmo em língua Portuguesa: Uma abordagem usando deep learning*: Universidade Federal de Mato Grosso. Tese de Mestrado. 10.13140/RG.2.2.18896.81924

Silva, Maurício. 2008. *O novo acordo ortográfico da língua portuguesa: o que muda, o que não muda*. Editora Contexto

Souza, Fábio, Rodrigo Nogueira & Roberto Lotufo. 2020. BERTimbau: Pretrained BERT models for Brazilian Portuguese. Em *Intelligent Systems*, 403–417. 10.1007/978-3-030-61377-8_28

Tavares, Diogo, Pedro Azevedo, David Semedo, Ricardo Sousa & João Magalhães. 2023. Task conditioned BERT for joint intent detection and slot-filling. Em *EPIA Conference on Artificial Intelligence*, 467–480. 10.1007/978-3-031-49008-8_37

Williams, Jason D, Antoine Raux & Matthew Henderson. 2016. The dialog state tracking

challenge series: A review. *Dialogue & Discourse* 7(3). 4–33. /10.5087/dad.2016.301

Wu, Shijie & Mark Dredze. 2020. Are all languages created equal in multilingual BERT? ArXiv [cs.CL]. 10.48550/arXiv.2005.09093

Zang, Xiaoxue, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang & Jindong Chen. 2020. MultiWOZ 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. Em *2nd Workshop on Natural Language Processing for Conversational AI*, 109–117. 10.18653/v1/2020.nlp4convai-1.13

Zhang, Jian-Guo, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wan, Philip S Yu, Richard Socher & Caiming Xiong. 2019. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. ArXiv [cs.CL]. 10.48550/arXiv.1910.03544

Zhang, Zheng, Ryuichi Takanobu, Qi Zhu, MinLie Huang & XiaoYan Zhu. 2020. Recent advances and challenges in task-oriented dialog systems. *Science China Technological Sciences* 63(10). 2011–2027. 10.1007/s11431-020-1692-3

Proposta recebida em Setembro 2024 e aceite para publicação em Novembro 2024.

Explorando Técnicas de Aprendizado em Modelos de Linguagem para Classificação de Discurso de Ódio e Ofensivo em Português

Exploring Learning Techniques in Language Models for Classifying Hate and Offensive Speech in Portuguese

Gabriel Assis
Universidade Federal Fluminense

Annie Amorim
Universidade Federal Fluminense

Jonnathan Carvalho
Instituto Federal Fluminense

Mariza Ferro
Universidade Federal Fluminense

Daniel de Oliveira
Universidade Federal Fluminense

Daniela Vianna
JusBrasil

Aline Paes
Universidade Federal Fluminense

Resumo

As Redes Sociais, que desempenham um papel significativo no debate e na comunicação moderna, enfrentam o desafio contemporâneo do grande volume desordenado de conteúdo nocivo, como discurso de ódio e desinformação. Este artigo aborda a detecção de discurso de ódio em português, considerando suas particularidades linguísticas e nuances culturais. Utilizando-se modelos derivados de *Transformers*, juntamente com diversas estratégias de treinamento e ativação, são investigados nove modelos com variações em arquitetura, tamanho e *corpora* de pré-treinamento. Os resultados obtidos demonstram que, apesar de grandes modelos generativos acessados via *prompts* apresentarem resultados promissores, modelos de linguagem de menor escala ajustados permanecem superiores na realização dessa delicada tarefa. ⚠ **O texto contém exemplos potencialmente nocivos e ofensivos.**

Palavras chave

transformers; classificação; discurso de ódio

Abstract

Social Media platforms, significant in modern debate and communication, face the challenge of managing a vast and disorderly volume of hateful content and disinformation. This work examines the detection of hate speech in Portuguese, contemplating its unique linguistic and cultural nuance. Leveraging Transformer-based models and different training and activation strategies, nine models with variations in architecture, size, and pre-training *corpora* are evaluated. Our findings show that, even though large generative models with enhanced prompts exhibited promising results, tuned small language models remain superior in addressing this task.

⚠ **The text contains potentially harmful and offensive examples.**

Keywords

transformers; classification; hate speech

1. Introdução

As redes sociais são um canal preeminente para disseminação de informações com uma velocidade sem precedentes, aumentando significativamente o alcance e a capacidade de comunicação e expressão de opiniões (Pelle et al., 2018). Tais plataformas evoluíram para praças públicas de debate, onde indivíduos e grupos podem compartilhar seus pontos de vista sobre uma ampla gama de tópicos (Moura, 2016; Paiva et al., 2019). No entanto, essas plataformas também amplificam problemas sociais, como a propagação de informações falsas e a proliferação de insultos e discursos de ódio (Aluru et al., 2020). Nesse contexto, comentários ofensivos são definidos como *aqueles que contêm qualquer tipo de comunicação ofensiva, transpassando desde linguagem inapropriada até insultos diretos* (Pelle et al., 2018). Por outro lado, o discurso de ódio é caracterizado como *qualquer expressão pública de ódio ou incentivo à violência contra um indivíduo, ou um grupo, baseados em características como etnia, raça, nacionalidade, orientação sexual e gênero* (Vargas et al., 2021). Essas expressões, quando endossadas, potencialmente resultam em ameaças à integridade individual e coletiva, emergindo assim como uma preocupação essencial para comunidades digitais, plataformas de mídia social, entidades governamentais e a sociedade como um todo (Saraiva et al., 2021).

DOI: 10.21814/lm.16.2.446
This work is Licensed under a Creative Commons Attribution 4.0 License

Adicionalmente, momentos de impacto expressivo no debate público podem tornar esse cenário ainda mais desafiador. A título de exemplo, nas eleições dos Estados Unidos em 2016, registrou-se um aumento nos crimes de ódio (Edwards & Rushin, 2018). Um efeito semelhante foi observado nas eleições federais brasileiras de 2018, quando houve um aumento massivo nos relatos de xenofobia, homofobia, racismo e intolerância religiosa nas mídias sociais (Vargas et al., 2021). Países como Canadá[1], Dinamarca[2] e Brasil[3] possuem arcabouços legais específicos para combater conteúdo de ódio em suas legislações. Especificamente na constituição brasileira, a discriminação baseada em raça, cor, etnia, religião ou origem nacional é legalmente reconhecida como crime. Contudo, certos usuários utilizam indevidamente as plataformas digitais para disseminar tal conteúdo, alicerçando-se erroneamente na prerrogativa de liberdade de expressão. Embora a liberdade de expressão também seja um direito constitucional, ela não deve promover ódio ou intolerância (de Freitas & de Castro, 2013). Não obstante, aplicar a lei permanece um desafio, principalmente devido ao volume de postagens e à complexidade de identificar e classificar corretamente comentários abusivos (Vargas et al., 2021). A análise precisa desse tipo de conteúdo é crucial, pois enfrentar o discurso de ódio transcende identificar comentários com linguagem abusiva, por exemplo. Apesar de plataformas digitais implementarem seus próprios sistemas de prevenção, eles apresentam várias limitações. Como ilustração, filtros sobre palavras-chave[4] podem endereçar o uso de expressões chulas, mas não nuances na expressão de ódio (Yin & Zubiaga, 2021). Além disso, muitos usuários empregam táticas inventivas ao escrever comentários ofensivos, como a troca de caracteres. Dessa forma, é decisivo construir métodos automatizados e precisos para filtrar e detectar conteúdos de discurso ofensivo e de ódio.

Este artigo concentra-se no contexto brasileiro, almejando contemplar as particularidades culturais do país e da língua portuguesa. A utilização contextual de expressões e palavras pode modificar profundamente o significado de um discurso, tornando fundamental uma abordagem que considere essas variações. Ilustrativamente, as sentenças ⚠ "*Ainda bem que gay não se reproduz..*", "*@USER canalha!*" e "*Olha que lindo!*" são classificadas nas bases de dados analisadas neste estudo como discurso de ódio, ofensivo e neutro, respectivamente. Enquanto a primeira sentença apresenta um indicativo claro de homofobia, a segunda representa uma ofensa direta, e a terceira reflete uma afirmação positiva, características que minimizam possíveis ambiguidades quanto à interpretação. Contudo, essa distinção nem sempre é evidente. Por exemplo, na sentença "*comecei a lavar a louça pra bichinha [...]*", originalmente rotulada como neutra, o uso do termo "bichinha"[5] pode gerar interpretações diversas, dependendo do leitor e do contexto. Em algumas regiões, a expressão é utilizada como vocativo informal entre interlocutores com quem se tem intimidade. No entanto, a comparação implícita com um animal pode ser considerada ofensiva por determinados receptores. Além disso, o termo pode carregar conotações homofóbicas, dependendo da intenção subjacente e do contexto em que é usado. Desse modo, fica evidente que a distinção das classes não é trivial, embora seja essencial. Os erros de classificação tornam-se particularmente críticos nesse contexto. Potenciais falsos positivos de discurso de ódio podem levar à censura indevida e, igualmente relevante, falsos negativos em tais classificações podem falhar em proteger grupos vulneráveis e comprometer a execução de leis.

Nesse ínterim, a arquitetura *Transformer* (Vaswani et al., 2017) emergiu, demonstrando resultados que compreendem o estado-da-arte em vários cenários, incluindo problemas de classificação e geração de texto (Fortuna & Nunes, 2018; Fu et al., 2024). Classificar postagens com origem em redes sociais é um campo de pesquisa ativo em Processamento de Linguagem Natural (PLN) (Fortuna & Nunes, 2018; Paiva et al., 2019; Jahan & Oussalah, 2023). Nessa perspectiva, enquanto o mundo se fascina pelas habilidades notáveis de grandes modelos de linguagem (*Large Language Models*, LLMs), como o ChatGPT[6], derivados dessa proeminente arquitetura, sobrepujar tarefas especialmente desafiadoras, como a identificação de discurso de ódio, permanece.

Ainda que utilizar LLMs para essa tarefa seja uma opção, ajustá-los é por vezes impraticável devido ao seu enorme número de parâmetros (sobretudo quando na ordem de centenas de bilhões) e a consequente implicação de custos diretos e indiretos. Nos casos de modelos de código-fechado, o acesso costuma ser realizado por meio de APIs[7] que resultam em cobranças associadas

[1] https://bit.ly/canadian-hate-speech-law
[2] https://bit.ly/danish-hate-speech-law
[3] https://bit.ly/planalto-lei-7716
[4] https://bit.ly/words-filter-threads-instagram

[5] Uso do termo "bichinho(a)": https://pt.m.wiktionary.org/wiki/bichim
[6] https://chat.openai.com/
[7] Interface de programação de aplicações, do inglês *Ap-*

ao seu uso direto. Contudo, mesmo nos modelos de código-aberto, emergem os custos indiretos decorrentes da necessidade de *hardware* capaz de comportar tais modelos. Dentre essas, uma possibilidade, ainda que não isenta de custos, porém mais viável do que ajustar os pesos do modelo, é o emprego do aprendizado contextualizado (*in-context learning*) (Brown et al., 2020). Nesse método, demonstrações são inseridas diretamente nos *prompts*—instruções textuais dadas ao modelo—para fornecer contexto relevante (Chiu et al., 2022). Por exemplo, ao instruir um modelo com um comando como *"Qual o sentimento em 'o filme é péssimo', dado que a frase 'a música é maravilhosa' tem sentimento positivo?"*, o modelo utiliza a demonstração fornecida para capturar como deve analisar e classificar a frase alvo, considerando o novo contexto dado sem a necessidade de ajustar diretamente os seus pesos.

Neste artigo, são avaliados diversos métodos de seleção de demonstrações: *one-shot* (que usa um único exemplo, independentemente da classe), *one-class-shot* (com um exemplo de cada classe) e *few-shot* (que utiliza mais de um exemplo para cada classe). Para selecionar exemplos de demonstração, é proposto escolhê-los com base em seu tamanho e proximidade de similaridade com as instâncias de teste. Essas estratégias são comparadas com a seleção de exemplos aleatórios e a não seleção de algum exemplo de demonstração (*zero-shot*). Outrossim, a adição de contexto extra no *prompt* por meio de palavras-chave selecionadas com técnicas de modelagem de tópicos (Amorim et al., 2023; Pham et al., 2024), enquanto a instrução é mantida fixa, também é experimentada.

Todavia, desponta a questão de se, mesmo com *prompts* aprimorados, LLMs estão preparados para lidar com as nuances específicas para identificar o discurso de ódio considerando-se apenas o aprendizado de seu pré-treinamento. Investiga-se também como classificadores baseados em modelos menores e ajustados se comparam aos LLMs mais recentes. Selecionaram-se modelos baseados em *encoder* e em *decoder*. Enquanto o primeiro grupo possui reconhecida eficácia em classificação (Fortuna & Nunes, 2018), o segundo demonstra capacidades em alinhar o significado semântico de rótulos com o texto de entrada (Li et al., 2023). Mesmo que ajustar esses modelos seja mais factível, outros fatores devem ser considerados, como as características dos *corpora* de pré-treinamento, contemplando estilo e comprimento do texto.

plication Programming Interface

Assim, a tarefa abordada neste artigo é formulada da seguinte maneira: **Dada uma postagem de rede social P escrita em português, realize o pré-processamento retornando X e classifique-o como pertencente a uma das classes em Y = {"discurso de ódio", "ofensivo" ou "neutro"}**. Em síntese, para essa tarefa de classificação, procedeu-se sobre três conjuntos de dados rotulados — HateBR (Vargas et al., 2022), OLID-BR (Trajano et al., 2023) e ToLD-Br (Leite et al., 2020) — à análise de nove modelos distintos, os quais se diferenciam quanto a sua arquitetura, tamanho e aos *corpora* em que foram pré-treinados. De forma específica, os modelos adotados podem ser categorizados em três grupos: **(i)** Modelos *encoder* fundamentados na arquitetura BERT (Devlin et al., 2019), o que neste trabalho incluem quatro variantes. Destacam-se, nesse grupo, três modelos especializados para a língua portuguesa, nomeadamente o BERTimbau (Souza et al., 2020), o AlBERTina (Rodrigues et al., 2023) e o BERTweet.BR (Caneiro et al., 2024) — esse último, um modelo pré-treinado especificamente em um *corpus* composto por *tweets*. Ademais, uma alternativa multilíngue, também pré-treinada em *tweets*, denominada Bernice (DeLucia et al., 2022); **(ii)** Modelos *decoder* ajustados para o português baseados na arquitetura LLaMA (Touvron et al., 2023a,b), compreendendo dois modelos de 7 bilhões de parâmetros — Sabiá (Pires et al., 2023) e Gervásio (Santos et al., 2024) — pré-treinados em textos formais; e, por fim, **(iii)** Grandes modelos de linguagem de propósito geral, dentre os quais os populares GPT (Brown et al., 2020) e Gemini (Google, 2023), e a MariTalk[8], um agente conversacional brasileiro desenvolvido especificamente para o português.

Substancialmente, são apresentadas quatro questões de pesquisa principais a respeito da classificação de postagens em redes sociais como neutras, ofensivas ou de discurso de ódio, em três conjuntos de dados:

QP1. Qual o desempenho de ajustar classificadores *encoder* clássicos a partir de modelos de linguagem "pequenos" em uma tarefa de classificação ternária e em um domínio delicado como a classificação de conteúdo de ódio?

QP2. A capacidade de alinhamento semântico de classificadores baseados em *decoder* os colocam como uma alternativa viável para a classificação de conteúdo de ódio?

[8] https://chat.maritaca.ai/

QP3. Adicionar contexto extra e demonstrações selecionadas de maneira acurada aprimora a resposta de grandes modelos de linguagem generativos acessados via *prompt*?

E, finalmente,

QP4. Modelos da arquitetura *Transformer* se mostram alternativas viáveis para a tarefa de classificação ternária de conteúdo de ódio em redes sociais?

No que concerne aos principais achados e contribuições deste trabalho, elencam-se os seguintes itens:

- Realizar o ajuste fino (*fine-tuning*) em um modelo de menor escala pré-treinado em *corpora* adequados impera neste domínio.
- Abordagens alicerçadas na capacidade de representação semântica de grandes modelos adotados como classificadores não prevalecem sobre os modelos clássicos de menor escala neste domínio. Isso sugere uma possível lacuna no pré-treinamento desse grupo de modelos sobre conteúdo odioso e ofensivo.
- Adicionar contexto e exemplos bem selecionados beneficia modelos generativos ativados por *prompt*. Desse modo, este artigo contribui com novas estratégias para o aprimoramento de *prompts* que podem ser investigadas em outros domínios e tarefas.

Destacamos que este artigo é uma versão estendida do estudo intitulado *"Exploring Portuguese Hate Speech Detection in Low-Resource Settings: Lightly Tuning Encoder Models or In-Context Learning of Large Models?"*, publicado nos anais da *16th International Conference on Computational Processing of Portuguese* (Assis et al., 2024). Nesta versão, novos experimentos foram realizados utilizando o conjunto de dados OLID-BR e novos modelos, Bernice e Gemini, foram avaliados. Além disso, aplicamos uma nova abordagem com classificadores baseados em *decoders*, Gervásio e Sabiá-1, e aprofundamos a análise qualitativa dos resultados. Também abordamos uma limitação do trabalho anterior relativa à subamostragem no conjunto de testes, proporcionando uma investigação mais abrangente e precisa do tema.

Por fim, o trabalho se encontra organizado em cinco seções além da Introdução. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve os modelos selecionados e as estratégias de aprendizado aplicadas. A Seção 4 detalha os conjuntos de dados utilizados. A Seção 5 discute os resultados experimentais. Por último, a Seção 6 conclui o presente artigo e apresenta trabalhos futuros.

O código desta investigação está disponível publicamente no GitHub[9].

2. Trabalhos Relacionados

Não obstante a identificação de discurso de ódio em redes sociais representar um tópico imperativo contemporaneamente, o número de estudos considerando as particularidades da língua portuguesa permanece limitado, sobretudo, se comparado ao Inglês (Trajano et al., 2023; Jahan & Oussalah, 2023). Todavia, alguns trabalhos aplicaram e investigaram classificadores tradicionais de aprendizado de máquina (Souza et al., 2022; Plath et al., 2022; Pelle et al., 2018; Silva et al., 2018; da Silva & Rosa, 2023; Paiva et al., 2019; Vargas et al., 2021, 2022), classificadores baseados em *Transformers* (da Silva & Rosa, 2023; Leite et al., 2020; Oliveira et al., 2023; Plath et al., 2022; Santos et al., 2022; Vargas et al., 2021) e grandes modelos generativos para abordar essa questão (Chiu et al., 2022; Das et al., 2023; Nguyen et al., 2023; Oliveira et al., 2023, 2024; Assis et al., 2024).

Em conjunturas específicas, como associadas à detecção de racismo, misoginia e homofobia, classificadores baseados em algoritmos como Naïve Bayes (NB), Máquinas de Vetores de Suporte (*Support Vector Machines*, SVMs) e Florestas Aleatórias (*Random Forests*, RFs) demonstram desempenhos preditivos expressivos (Souza et al., 2022; Plath et al., 2022; Silva et al., 2018). Outrossim, alguns trabalhos utilizam representações baseadas em *embeddings* (Pelle et al., 2018; da Silva & Rosa, 2023). Contudo, essa abordagem pode apresentar limitações na representação de palavras sensíveis a contextos dinâmicos, como em redes sociais (Seno et al., 2024).

Modelos baseados em BERT (Devlin et al., 2019) despontam como o estado-da-arte proeminente na classificação de discurso de ódio, com alguns modelos específicos para línguas superando alternativas multilíngues em contextos não anglófonos (Jahan & Oussalah, 2023). Nesse sentido, da Silva & Rosa (2023) avaliaram 11 métodos distintos de classificação, incluindo o BERTimbau (Souza et al., 2020), que alcançou os melhores resultados para a língua portuguesa. De maneira similar, outros trabalhos destacaram o desempenho superior do BERTimbau (da Silva & Rosa, 2023; Santos et al.,

[9] https://github.com/MeLLL-UFF/hate_speech_in_context_pt

2022; de Souza et al., 2024) e do BERT multilíngue (Leite et al., 2020) sobre abordagens que utilizam representações construídas sobre extração de *embeddings*, entretanto considerando a detecção de discurso de ódio apenas como um problema de classificação binário.

Para mais, os LLMs e suas notáveis habilidades emergentes também têm sido avaliados na tarefa de detecção de texto ofensivo e discurso de ódio. Chiu et al. (2022) investigaram as capacidades do ChatGPT para a detecção de linguagem sexista e racista, fazendo uso de técnicas de aprendizado *zero-shot*, *one-shot* e *few-shot*. Em contraste, Oliveira et al. (2023) utilizaram-se exclusivamente da abordagem *zero-shot* para avaliar o desempenho do GPT na detecção de discurso de ódio em *tweets* em português. Nesse trabalho, a comparação com o BERTimbau ajustado demonstra a viabilidade de modelos generativos para classificar conteúdo odioso. Em (Das et al., 2023), o ChatGPT demonstrou resultados promissores na detecção de discurso de ódio em português, porém com limitações sobre a capacidade de distinção entre discurso abusivo e não odioso dirigido a indivíduos e minorias. Ainda, Nguyen et al. (2023) avaliaram modelos LLaMA-2 ajustados para a detecção de textos sexuais, predatórios e abusivos. Por fim, os trabalhos de ambos, Assis et al. (2024) e Oliveira et al. (2024), contrastaram o GPT-3.5 e o modelo brasileiro MariTalk com alternativas baseadas em BERT ajustadas para o português brasileiro, concluindo que o último grupo alcança melhor desempenho, mesmo com resultados promissores das alternativas generativas.

Nenhum dos trabalhos mencionados conduziu um estudo que incluísse, de forma comparativa, a mesma vasta quantidade de modelos em língua portuguesa ajustados para tratar de um problema de classificação ternária sobre este domínio. As pesquisas anteriores concentram-se em problemas de classificação com configurações distintas da adotada neste trabalho. Por exemplo, Trajano et al. (2023) reporta métricas gerais de F1 na ordem de 0,70 para a tarefa de classificação multiclasse no conjunto OLID-BR, abrangendo 10 rótulos, incluindo racismo, sexismo e xenofobia. Observa-se, entretanto, que os resultados variam significativamente entre as classes. As classes de xenofobia e racismo, por exemplo, que possuem um suporte inferior a 40 amostras, apresentam métrica de F1 inferiores a 0,50. No caso da configuração binária, em que se considera uma classe neutra e uma contra-classe tóxica, Oliveira et al. (2024) reporta resultados de F1 de 0,75 para a classe de ódio no conjunto ToLD-BR, enquanto de Souza et al. (2024) alcança métricas na ordem de 0,90 no conjunto HateBR. Neste estudo, optamos pela abordagem de classificação ternária, o que restringe comparações diretas com os resultados mencionados. Nesse sentido, essa escolha busca proporcionar uma análise mais granular das classes, ampliando o escopo de investigação para além das configurações binárias frequentemente exploradas. Ao mesmo tempo, busca-se equilibrar a representação dos rótulos, evitando a formação de classes extremamente subamostradas e que podem ter seus resultados sub-representados. De mais a mais, da mesma maneira, modelos baseados em *decoder* como base para classificadores e um LLM mais recente em uma abordagem de aprendizado contextualizado não foram também avaliados. Além disso, este trabalho introduz novas estratégias para a seleção de demonstrações em *prompts* e propõe um método baseado em modelagem de tópicos para enriquecer instruções.

3. Método

Esta seção descreve os modelos selecionados, suas características, os métodos de treinamento adotados e as estratégias de inferência propostas.

3.1. Modelos

Para responder às questões de pesquisa QP1 e QP2, que abordam o desempenho de classificadores baseados em arquiteturas de *encoder* e *decoder*, respectivamente, bem como a QP4, que trata do desempenho geral de modelos *transformers* no domínio da classificação de discurso de ódio, foi realizada uma seleção criteriosa de modelos. Essa seleção incluiu explicitamente tanto arquiteturas baseadas em *encoders* quanto em *decoders*, além de grandes modelos de linguagem acessados por meio de *prompts*. Primeiro, destacam-se os modelos baseados em BERT (Devlin et al., 2019), pré-treinados com *corpora* específicos da variante brasileira do português: (i.) BERTimbau (Souza et al., 2020), em sua versão "*large*", e (ii.) AlBERTina PT-BR (Rodrigues et al., 2023), na sua versão de 100 milhões de parâmetros, ambos pré-treinados com textos mais formalmente redigidos. Além desses, (iii.) BERTweet.BR (Caneiro et al., 2024), pré-treinado com um *corpus* de *tweets* brasileiros, e (iv.) Bernice (DeLucia et al., 2022), que, embora multilíngue, também foi pré-treinado com dados do Twitter[10].

[10] A plataforma possui atualmente o nome X: `https://x.com/`

No que tange aos classificadores baseados em *decoder*, selecionam-se (v.) Sabiá-7B-1 (Pires et al., 2023), ajustado sobre a arquitetura LLaMA-1 (Touvron et al., 2023a), e (vi.) Gervásio-7B-PTBR (Santos et al., 2024), baseado na arquitetura LLaMA-2 (Touvron et al., 2023b). Ambos os modelos foram pré-treinados com textos formais e orientados para tarefas de geração de conteúdo autorregressivo.

Quanto aos LLMs acessados via *prompt*, incluem-se (vii.) GPT-3.5-turbo (Ouyang et al., 2022), da reconhecida família GPT; (viii.) Gemini-pro 1.0 (Google, 2023), modelo notável por seus resultados expressivos em *benchmarks* recentes; e (ix.) MariTalk, um agente brasileiro aplicado na sua versão construída sobre o modelo Sabiá-2-medium (Almeida et al., 2024), ajustado sobre um *corpus* em português para também realizar tarefas gerais de um *chatbot*.

Considerando os diversos tamanhos, a natureza e a disponibilidade desses modelos, são aplicadas estratégias distintas para cada grupo. No entanto, enfatiza-se que o desempenho preditivo é avaliado consistentemente sobre o mesmo conjunto de testes. Detalhamentos adicionais serão apresentados nas seções subsequentes.

3.2. Treinamento dos classificadores

O processo de ajuste de pesos de classificadores permite que o modelo preserve as representações linguísticas obtidas no pre-treinamento, todavia se adequando a novos padrões de um novo domínio ou tarefas específicas, melhorando não só o seu desempenho nas previsões, mas também a sua capacidade de generalização (Yosinski et al., 2014). Desse modo, as técnicas empregadas para o ajuste dos modelos classificadores usados neste trabalho são especificadas a seguir.

3.2.1. Classificadores Baseados em Encoder

Os modelos de *encoder* foram ajustados utilizando duas abordagens principais: extração de características (*feature extraction*) e ajuste fino (*fine-tuning*). Apesar de o ajuste fino geralmente oferecer um melhor desempenho preditivo, a extração de características também foi explorada neste trabalho, considerando a sua adoção em diversos trabalhos anteriores no domínio da detecção de discurso de ódio (Fortuna et al., 2019; Plath et al., 2022).

Neste estudo, a abordagem de extração de características utiliza o *token* [CLS] como base para as representações vetoriais de entrada para treinar classificadores SVM. Esse *token* é um marcador especial inserido no início das sequências de entrada em modelos de linguagem. Ele desempenha a função de capturar a essência da sequência inteira, uma vez que agrega um resumo contextual de todas as entradas posteriores. Dessa forma, sua representação vetorial é frequentemente utilizada em tarefas de classificação que envolvem sentenças. Os vetores de características são então extraídos dos modelos de linguagem, originando a matriz $\mathbf{X} \in \mathrm{R}^{n \times d}$, onde $\mathbf{X}$ representa a matriz de exemplos, n o número de exemplos, e d as dimensões correspondentes ao *token* [CLS]. Assim, apenas os parâmetros do novo classificador são ajustados com base no conjunto de dados de treinamento, mantendo-se os pesos do modelo de linguagem pré-treinado inalterados. A outra estratégia adotada é a do ajuste fino, que consiste em acoplar uma camada classificatória diretamente ao modelo de linguagem. Por fim, os modelos são então inteira e diretamente ajustados conforme os exemplos do conjunto de treinamento.

3.2.2. Classificadores Baseados em Decoder

Modelos baseados em *decoders* também são adaptáveis a tarefas específicas, incluindo contextos que envolvem conteúdo sensível (Nguyen et al., 2023). No entanto, a estratégia de ajuste comumente utilizada aproveita a tarefa original de pré-treinamento desses modelos, que, geralmente, é a geração de texto de forma autorregressiva. Esse método de ajuste, embora permita a adaptação dos modelos a novos domínios e instruções, muitas vezes não utiliza diretamente conjuntos de dados rotulados, mesmo quando esses estão disponíveis. Para abordar esse quesito, uma estratégia semelhante à utilizada em classificadores baseados em *encoders* pode ser aplicada: incorporar uma camada classificatória diretamente ao modelo de linguagem, ao encontro do processo de ajuste fino. Essa abordagem se baseia no conceito de que as saídas dos modelos baseados em *decoders* podem alinhar o significado semântico da entrada com os rótulos, podendo, pois, funcionar como representações textuais para tarefas de classificação, inclusive alcançando resultados notáveis (Li et al., 2023).

3.3. Ativação dos Modelos Generativos via Prompts

As respostas obtidas dos agentes GPT-3.5-turbo[11] e MariTalk[12] são coletadas das suas res-

[11] https://platform.openai.com/
[12] https://www.maritaca.ai/

pectivas APIs. Já o Gemini-pro é acessado por meio do serviço da Google Cloud Platform, Vertex AI[13]. Esses agentes recebem como entrada uma instrução que define a tarefa a ser realizada, um contexto que incorpora informações adicionais relevantes, e um ou mais exemplos de demonstração, que são pares de entrada e saída (X, Y_i) utilizados como referência. Neste estudo, são abordadas três classes, portanto, Y_i pode ser neutro, ofensivo ou discurso de ódio. São propostas múltiplas maneiras de selecionar exemplos de demonstração. Adicionalmente, também são experimentados diferentes contextos. A instrução permanece inalterada ao longo dos experimentos. Embora se entenda a sensibilidade desses agentes em relação as suas entradas (Liu et al., 2023), estudos anteriores que exploraram previamente instruções para a detecção de discurso de ódio em português são considerados como alicerce (Oliveira et al., 2023). Complementarmente, almeja-se, sobretudo, investigar o papel do contexto e das demonstrações na composição dos *prompts*, considerando a questão de pesquisa QP3, que aborda os efeitos da inclusão de informações contextuais adicionais e exemplos sistematicamente selecionados.

3.3.1. Construção do Prompt

Duas principais fontes direcionam a instrução aplicada neste trabalho. A primeira é o PromptHub[14], um repositório de código aberto de *prompts* categorizados por tarefa. Os *prompts* dessa coleção relacionados a tarefas semelhantes, como análise de sentimentos, auxiliaram a moldar a formulação da instrução. Em contrapartida, Pires et al. (2023) embasaram a integração de demonstrações dentro dos *prompts*. Assim, define-se a seguinte instrução[15]: Classifique o texto de rede social como "discurso de ódio" ou "ofensivo" ou "neutro".\n Texto: ≪ *alvo* ≫ \n Classe:.

Especificamente para o MariTalk, a instrução para ativação do modelo foi complementada com a sentença "Responda apenas com a classe". Isso se deve ao fato de que, em testes preliminares, observou-se que esse modelo tende a ser verboso em suas respostas, por vezes elaborando justificativas para suas classificações. Embora essa característica possa ser vantajosa em certos contextos, como na busca por explicabilidade, a crescente no número de *tokens* e o consequente aumento nos custos podem ser limitantes, considerando-se, sobretudo, o grande volume de experimentos realizados neste trabalho. Dessa forma, a adição dessa diretriz permitiu que o modelo realizasse classificações de maneira concisa e com respostas autocontidas no número de *tokens* estabelecido.

3.3.2. Estratégias para Seleção de Demonstrações

Sobre o número de exemplos de demonstração, foram aplicadas quatro maneiras de compor os *prompts*: ***(a.) zero-shot***, caracterizado pela ausência de exemplos no comando; ***(b.) one-shot***, que inclui um único exemplo, sem distinção de classe; ***(c.) one-class-shot***, que engloba um exemplo para cada classe no comando; e ***(d.) few-shot***, que apresenta vários exemplos para cada classe. Esses exemplos são selecionados a partir do conjunto de dados de treinamento.

Para a seleção de demonstrações do conjunto de treinamento, foram propostas três estratégias visando reduzir a variabilidade e aumentar a eficácia. A primeira estratégia consiste em ***(e.)*** **selecionar exemplos aleatoriamente**, considerando, naturalmente, a quantidade estabelecida de exemplos de demonstração. Por exemplo, essa abordagem, combinada com a modalidade (c.), implica a escolha aleatória de um exemplo de cada classe; já na modalidade (b.), envolve a seleção de um único exemplo de todo o conjunto de treinamento. Outras duas estratégias levam em conta ***(f.)*** **a similaridade semântica**, baseando-se nas representações de *embeddings*, ou ***(g.)*** **o tamanho em número de *tokens***, para a escolha dos exemplos. Intuitivamente, pretende-se fornecer informações adicionais, porém relevantes, para orientar melhor a capacidade de aprendizado contextualizado.

Ambas as últimas estratégias iniciam-se com a criação de agrupamentos

$$C = \{C_{1,1}, \ldots, C_{1,k_1}, C_{2,1}, \ldots, C_{2,k_2}, C_{3,1}, \ldots, C_{3,k_3}\},$$

formados separadamente para cada uma das três classes, visando assegurar uma clara distinção. Pressupõe-se que todas as instâncias de teste se enquadrarão no mesmo agrupamento C_t, uma vez que sua classe é desconhecida em tempo de execução. O passo seguinte é identificar, um para cada classe, os agrupamentos mais próximos $C_{1,p}, C_{2,q}$ e $C_{3,r} \in C$ e os agrupamentos $C_{1,s}, C_{2,t}$ e $C_{3,u} \in C$ mais distantes das representações de *embeddings* médias das instâncias de teste C_t, visando compreender como esses extremos podem influenciar positiva ou negativamente o aprendizado. Destaca-se que, embora o conjunto de testes seja utilizado para formar C_t, somente

[13] https://cloud.google.com/vertex-ai/
[14] https://github.com/deepset-ai/prompthub/
[15] Um exemplo completo de um *prompt one-shot* encontra-se no Apêndice A.

sua informação textual é adotada, o que viabiliza a aplicabilidade da estratégia em tempo de execução.

Na análise do impacto das informações extremas, a **estratégia baseada na similaridade semântica (f.)** seleciona (f.1.) os exemplos

$$\text{Ex} = \{ex_{x_1} \in C_{1,p},\ ex_{x_2} \in C_{2,q},\ ex_{x_3} \in C_{3,r}\}$$

que estão mais próximos dos *embeddings* médios de C_t, conforme a similaridade de cosseno, ou, alternativamente, (f.2.) os exemplos

$$\text{Ex} = \{ex_{z_1} \in C_{1,s},\ ex_{z_2} \in C_{2,t},\ ex_{z_3} \in C_{3,u}\}$$

que se distanciam mais. Novamente, essa escolha deve alinhar-se às configurações de a-d, adaptando-se ao número de exemplos requeridos em cada caso. Por exemplo, a estratégia (d.) realiza o processo múltiplas vezes até selecionar N exemplos, enquanto a (c.), apenas uma. Já para a estratégia (b.), seleciona-se apenas um exemplo de cada classe em Ex por vez. A **estratégia baseada no tamanho (g.)** é construída sobre (f.). Todavia, prioriza os exemplos que, além de estarem semanticamente próximos ou distantes, apresentam um tamanho em número de *tokens* mais próximo à moda das instâncias de teste. Essa abordagem é fundamentada na percepção de que conteúdos semanticamente relacionados podem compartilhar uma quantidade de *tokens* similar, facilitando a transmissão de mensagens análogas.

3.3.3. Contexto Adicional

Foram abordadas duas estratégias distintas para investigar se há benefício em incluir mais contexto aos *prompts*: a primeira absteve-se do uso de contexto adicional, ao passo em que a segunda incorporou palavras-chave e *emojis*[16] para dotar os modelos de exemplos representativos do tipo de discurso em foco. A seleção dessas palavras-chave pode desempenhar um papel significativo em um contexto em que a anotação de dados manual é realizada sob orientação de classificação com base na existência de termos específicos (Vargas et al., 2022). O método proposto é articulado em quatro etapas. Inicialmente, procede-se à remoção de pronomes possessivos, nomes próprios, verbos, caracteres especiais, numerais e termos com menos de duas letras. Em um segundo momento, para cada classe definida, são gerados dez tópicos a partir do conjunto de dados de treinamento, utilizando-se o BERTopic (Grootendorst, 2022) em integração com o BERTweet.BR. Subsequentemente, computa-se o cálculo da frequência de palavras por classe, identificando-se as dez mais prevalentes. No último passo, selecionam-se as dez palavras mais pertinentes dos tópicos gerados de acordo com seus pesos, desde que não constem nos tópicos ou no conjunto de palavras frequentes de outras classes.

As palavras-chave são incorporadas ao *prompt*, situando-se entre a instrução descrita e as demonstrações selecionadas, conforme o seguinte formato[17]: Considerando que os assuntos da classe "Classe A" estão associados com as palavras e emojis ≪*10 termos mais relevantes para Classe A* ≫. \n Da classe "Classe B" estão associados com as palavras e emojis ≪*10 termos mais relevantes para Classe B* ≫. \n Da classe "Classe C" estão associados com as palavras e emojis ≪*10 termos mais relevantes para Classe C* ≫.

4. Conjuntos de Dados

Os modelos foram avaliados em três conjuntos de dados distintos: HateBR (Vargas et al., 2022), OLID-BR (Trajano et al., 2023) e ToLD-Br (Leite et al., 2020). O conjunto HateBR consiste em 7.000 comentários do Instagram[18], recolhidos de perfis de políticos brasileiros durante o segundo semestre de 2019. Esse conjunto foi dividido em três classes para esta pesquisa: discurso de ódio (abrangendo textos rotulados com ao menos uma das classes dentre misoginia, gordofobia, xenofobia), textos ofensivos (que não se encaixam em algum conteúdo de ódio relacionado à construção identitária) e textos neutros.

Por outro lado, o conjunto de dados OLID-BR contém 7.000 registros que remontam a 2019, provenientes de comentários no Youtube[19] e postagens no Twitter. De maneira semelhante, textos identificados com ao menos um dos rótulos associados a manifestações de LGBTQ+fobia, racismo, sexismo, intolerância religiosa, ou outras formas de insultos dirigidos a grupos identitários, foram agrupados na classe de discurso de ódio. Textos que, não estando diretamente ligados ao discurso de ódio, porém marcados em rótulos como insultos ou profanidades, constituem a classe de textos ofensivos neste conjunto. Textos que não se ajustam a qualquer uma das classes de ofensividade ou discurso de ódio são rotulados como neutros.

[16] https://www.significados.com.br/emoji/

[17] Um exemplo completo de um *prompt* com contexto encontra-se no Apêndice B.

[18] https://www.instagram.com/

[19] https://www.youtube.com/

O conjunto ToLD-Br, por sua vez, inclui 21.000 *tweets* coletados entre julho e agosto de 2019, rotulados como não tóxico, LGBTQ+fobia, obsceno, insulto, racismo, misoginia e xenofobia. É notável que apenas cerca de 300 *tweets* foram estritamente rotulados em um dos rótulos de discurso de ódio. Neste estudo, as mensagens categorizadas como obscenas ou insultuosas foram agrupadas na classe ofensiva, enquanto aquelas consideradas não tóxicas foram classificadas como neutras, e as demais foram incluídas na classe de discurso de ódio.

Outrossim, a classificação do HateBR foi conduzida por anotadores voluntários com formação de doutorado e especialização em linguística, discurso de ódio ou ciência da computação. Já o processo para o OLID-BR começou com um filtro inicial feito por modelos automáticos de detecção de toxicidade[20] e prosseguiu com anotação humana. Para serem considerados minimamente qualificados para a tarefa, os anotadores precisavam ser nativos ou fluentes em português, além de atenderem a critérios de formação específicos. O ToLD-Br, por sua vez, contou com pelo menos três anotadores por instância, sem impor restrições quanto à formação anterior desses. Em todos os conjuntos de dados, foi dada ênfase à diversidade de gênero, à orientação política e à diversidade racial entre os selecionados. Vale ressaltar que o OLID-BR foi o único conjunto de dados que remunerou seus anotadores.

Dados	N	O	D	#tokens
HateBR	3410	2670	695	15.83
OLID-BR	1016	3896	2029	28.39
ToLD-Br	11634	8241	290	18.78

Tabela 1: Número de instâncias por classe, denotados como **N**eutro, **O**fensivo e **D**iscurso de Ódio, e contagem média de *tokens* por instância nos conjunto de dados.

A Tabela 1 apresenta a proporção das classes, acompanhada do número médio de *tokens* por instância, calculado com base no modelo BERTweet.BR, dado o contexto de redes sociais. A contagem média reflete possíveis diferenças no comportamento dos usuários nas plataformas que originaram os conjuntos de dados (French & Bazarova, 2017). Por exemplo, o conjunto HateBR, apresenta a menor contagem média de *tokens*, o que pode estar relacionado ao padrão de interação no Instagram, em que as postagens textuais geralmente consistem em comentários breves e reativos às fotografias publicadas. Em contraste, o conjunto ToLD-BR possui uma contagem ligeiramente maior, potencialmente devido a sua origem no Twitter, uma rede predominantemente textual, porém com limitações no número de caracteres por postagem. Já o conjunto OLID-BR, por combinar dados de múltiplas fontes, dentre elas o YouTube, talvez represente um contexto no qual os usuários dispõem de mais espaço para elaborar comentários, opiniões e análises relacionadas a vídeos.

Enfim, todos os conjuntos de dados foram particionados em aproximadamente 60% para treinamento e 20% para validação e testes cada, mantendo a proporção das classes. Para alcançar o balanceamento, a subamostragem aleatória das classes majoritárias foi aplicada no conjunto de treinamento.

5. Resultados Experimentais

Esta seção apresenta as configurações usadas para os experimentos e discute o desempenho dos modelos e estratégias adotados, com foco na classe de discurso de ódio, conjecturada por nós como a mais crítica no domínio analisado. Em seguida, realiza-se uma análise comparativa dos desempenhos proeminentes nas demais classes, prosseguindo-se então para uma inspeção qualitativa sobre os resultados.

5.1. Configurações Experimentais

Realizou-se um procedimento de pré-processamento simples sobre os dados, envolvendo a eliminação de duplicidades, a substituição de menções a usuários pelo *token* @USER, de links por HTTPURL e de *emojis* por sua representação textual, por meio da biblioteca Emoji[21]. A determinação dos agrupamentos C, como parte das estratégias (f.) e (g.), referidas na Seção 3.3.2, fundamenta-se no método tradicional KMeans (Jin & Han, 2011). Estabelece-se o número de grupos consoante o Critério do Cotovelo (Thorndike, 1953), sendo $k = 4$ para todas as classes.

O ajuste de classificadores baseados em *encoder*, abordado na Seção 3.2.1, recorre a hiperparâmetros tipicamente aplicáveis a modelos desse tipo. Em um cenário de recursos limitados, os modelos foram calibrados adotando uma taxa de aprendizado de $2e-5$, um tamanho de lote de 16 e com uso do critério de parada antecipada igual a 3, almejando evitar o sobreajuste.

[20] https://www.perspectiveapi.com/

[21] https://pypi.org/project/emoji/

Dados	Modelo	acurácia	precisão	revocação	F1
HateBR	**BERTimbau (extração de características)**	0.756	0.361	0.633	0.460
	BERTimbau (ajuste fino)	**0.862**	0.632	**0.705**	**0.667**
	AlBERTina PT-BR (extração de características)	0.430	0.000	0.000	0.000
	AlBERTina PT-BR (ajuste fino)	0.800	**0.785**	0.446	0.569
	BERTweet.BR (extração de características)	0.292	0.115	0.612	0.194
	BERTweet.BR (ajuste fino)	0.846	0.529	0.647	0.583
	Bernice (extração de características)	0.725	0.312	0.698	0.431
	Bernice (ajuste fino)	0.805	0.750	0.583	0.656
OLID-BR	**BERTimbau (extração de características)**	0.581	0.600	0.533	0.565
	BERTimbau (ajuste fino)	0.663	0.729	0.504	0.596
	AlBERTina PT-BR (extração de características)	0.253	0.387	0.257	0.309
	AlBERTina PT-BR (ajuste fino)	0.613	**0.749**	0.427	0.544
	BERTweet.BR (extração de características)	0.525	0.475	**0.649**	0.548
	BERTweet.BR (ajuste fino)	**0.672**	0.625	0.588	**0.606**
	Bernice (extração de características)	0.558	0.646	0.501	0.565
	Bernice (ajuste fino)	0.666	0.737	0.477	0.579
ToLD-Br	**BERTimbau (extração de características)**	0.615	0.046	0.534	0.084
	BERTimbau (ajuste fino)	0.599	0.035	**0.569**	0.065
	AlBERTina PT-BR (extração de características)	0.546	0.000	0.000	0.000
	AlBERTina PT-BR (ajuste fino)	0.538	0.033	0.241	0.059
	BERTweet.BR (extração de características)	0.535	0.000	0.000	0.000
	BERTweet.BR (ajuste fino)	**0.708**	**0.105**	**0.569**	**0.178**
	Bernice (extração de características)	0.616	0.046	0.552	0.085
	Bernice (ajuste fino)	0.704	0.082	0.500	0.141

Tabela 2: Resultados preditivos de classificadores baseados em *encoders* nas estratégias de extração de características e ajuste fino. Exceto pela acurácia, os valores são computados para a classe de discurso de ódio. Valores em **negrito** representam as melhores métricas para cada conjunto de dados.

Já o ajuste para os classificadores baseados em *decoder*, tratado na Seção 3.2.2, ocorreu sobre a estratégia LoRA (Hu et al., 2022), com $r = 16$, $lora_alpha = 32$, e taxa de aprendizado $1e-4$. O tamanho de lote de 16 e o critério de parada antecipada também foram adotados.

Quanto ao grupo de experimentos realizados com modelos generativos acessados via *prompts*, fixou-se o limite máximo de *tokens* por resposta em 20, e a amostragem foi desabilitada. O ajuste do parâmetro de temperatura foi de 0.1 para todos os modelos, *i.e.*, GPT, Gemini-pro e MariTalk. Particularmente, é importante mencionar que o parâmetro $BLOCK_NONE$ foi utilizado nas configurações de segurança da API do Gemini-pro para atenuar os bloqueios decorrentes das políticas éticas do Google, devido à sensibilidade dos dados. Além disso, optou-se por configurar o parâmetro *chat_mode* do MariTalk como $True$, de encontro à recomendação da documentação oficial para desativá-lo[22]. Essa decisão foi tomada para evitar respostas insubstanciais observadas nos testes realizados no contexto deste trabalho, incluindo respostas compostas unicamente de caracteres vazios (“ ”) ou contendo somente o caractere indicador de quebra de linha (\n). Já para a estratégia (d.) *few-shot*, dois exemplos por classe foram adotados.

Por fim, implementaram-se os classificadores baseados em *encoder* e *decoder* utilizando como arcabouço o *framework transformers* da Hugging Face (Wolf et al., 2020) e com o uso de recursos em nuvem da Google Cloud Platform (GCP)[23]. Foram utilizadas instâncias com quatro GPUs T4. O Scikit-learn (Pedregosa et al., 2011) foi adotado para o treinamento das SVMs, no esquema *one-vs-one* (OvO).

5.2. Resultados dos Classificadores

5.2.1. Resultados dos Classificadores Baseados em Encoder

A fim de responder à primeira questão de pesquisa— *Qual o desempenho de ajustar classificadores encoder clássicos a partir de modelos de linguagem “pequenos” em uma tarefa de classificação ternária e em um domínio delicado como a classificação de conteúdo de ódio?* —, a Tabela 2 detalha os resultados em termos de precisão, revocação e medida F1 para a classe de discurso de ódio, além de apresentar o valor macro de acurácia.

BERTimbau e Bernice dividem os melhores resultados para a estratégia de extração de características. Esperava-se que o modelo BERTweet.BR, dada sua fase de pré-treinamento em um *corpus* de redes sociais e especificamente no português do Brasil, apresentasse desempenhos superiores. No entanto, uma presunção considerada é a de que esse modelo possa apresen-

[22] https://maritaca-ai.github.io/maritalk-api/maritalk.html

[23] https://cloud.google.com

Dados	Modelo	acurácia	precisão	revocação	F1
HateBR	**Sabiá-1-7B**	0.526	0.206	0.094	0.129
	Gervásio-7B-PTBR	**0.672**	**0.270**	**0.475**	**0.345**
OLID-BR	**Sabiá-1-7B**	**0.532**	0.442	0.385	0.412
	Gervásio-7B-PTBR	0.495	**0.465**	**0.486**	**0.475**
ToLD-Br	**Sabiá-1-7B**	**0.531**	0.000	0.000	0.000
	Gervásio-7B-PTBR	0.446	**0.023**	**0.259**	**0.042**

Tabela 3: Resultados preditivos de classificadores baseados em *decoders* na estratégia de ajuste fino. Exceto pela acurácia, os valores são computados para a classe de discurso de ódio. Valores em **negrito** representam as melhores métricas para cada conjunto de dados.

tar uma representação de vocabulário excessivamente especializada. Tal especificidade, sem algum ajuste, pode não ter sido eficaz em um procedimento de classificação de múltiplas classes. BERTimbau, por sua vez, foi pré-treinado em um *corpus* em português, mas não de redes sociais, enquanto Bernice foi pré-treinado em um *corpus* de *tweets*, mas multilíngue. Essas características podem conferir a ambos os modelos uma representação menos restritiva, que, sem a superespecialização, facilitou possivelmente a atuação da SVM na melhor distinção das diferentes classes.

Por outro lado, ao se considerar a estratégia de ajuste fino, o BERTweet.BR apresenta superioridade geral ao alcançar os melhores resultados na metade das métricas. Contudo, esses resultados são concentrados nos conjuntos OLID-BR e ToLD-BR, que incluem *tweets* em seus dados. Em contrapartida, o modelo BERTimbau, na configuração de ajuste fino, apresenta um desempenho superior no conjunto de dados HateBR, composto por comentários provenientes do Instagram. Essa observação pode estar associada à existência de comportamentos distintos entre os usuários dessas plataformas. Isso porque usuários tendem a escrever de maneira mais espontânea e informal em microblogs como o Twitter, enquanto adotam uma postura mais formal nos comentários do Instagram (French & Bazarova, 2017). Essa variação pode ter favorecido o BERTimbau nesse contexto.

5.2.2. Resultados dos Classificadores Baseados em Decoder

Associada à segunda questão de pesquisa— *A capacidade de alinhamento semântico de classificadores baseados em decoder os colocam como uma alternativa viável para a classificação de conteúdo de ódio?* —, a Tabela 3 dispõe os resultados dos modelos *decoders* adaptados como classificadores por meio do ajuste fino.

Apesar de possuírem mais pesos, os modelos classificadores baseados em *decoders* de 7 bilhões de parâmetros demonstraram ser menos eficazes quando comparados aos modelos fundamentados em *encoders*, consoante contraste da Tabela 2 e Tabela 3. Isso sugere uma possível lacuna no processo de pré-treinamento desses modelos no que tange ao conteúdo de ódio, uma limitação conjecturável dado os plausíveis filtros implementados para prevenir que esses modelos, tipicamente empregados como generativos, propaguem tal conteúdo. Adicionalmente, a configuração usual da pilha *decoder* implica um processamento unidirecional dos *tokens*, orientado para a emissão de símbolos. Tal configuração pode dificultar o seu uso direto, sem modificações, como classificadores. Por fim, a relação entre o tamanho dos conjuntos de dados e o número de parâmetros desses modelos, bem como a viabilidade de adaptá-los utilizando exclusivamente a estratégia LoRA, pode representar um percalço durante a fase de ajuste. De modo geral, o modelo Gervásio sobressaiu-se, apresentando os melhores desempenhos, com exceção da acurácia nos conjuntos OLID-BR e ToLD-BR. Nesse sentido, é relevante destacar que o Gervásio é um modelo de construção mais recente, baseado em uma arquitetura mais atual, LLaMA-2, em contraste com a versão avaliada do Sabiá, que utiliza a arquitetura LLaMA-1.

5.3. Resultados da Inferência com LLMs

As Tabelas 4, 5, 6 e 7 apresentam os resultados das inferências feitas com LLMs generativos acessados via *prompt*, variando as diversas estratégias de seleção de demonstração introduzidas na Seção 3.3.2. Dessa maneira, pretende-

Modelo	#exemplos	seleção	acurácia	(±%)	precisão	(±%)	revocação	(±%)	F1	(±%)
GPT-3.5-turbo	**zero-shot**	**-**	0.649	ref.	0.237	ref.	0.770	ref.	0.362	ref.
	one-shot	**aleatória**	0.697	+7%	0.287	+21%	0.604	-22%	0.389	+7%
		similaridade	0.691	+6%	0.256	+8%	0.748	-3%	0.381	+5%
		tamanho	0.694	+7%	0.291	+23%	0.676	-12%	0.407	+12%
	one-class-shot	**aleatória**	0.535	-18%	0.210	+11%	**0.906**	**+18%**	0.341	-6%
		similaridade	0.641	-1%	0.237	0%	0.856	+11%	0.371	+2%
		tamanho	0.697	+7%	0.278	+17%	0.763	-1%	**0.408**	**+13%**
	few-shot	**aleatória**	0.672	+4%	0.223	-6%	0.640	-17%	0.330	-9%
		similaridade	**0.751**	**+16%**	**0.306**	**+29%**	0.424	-45%	0.355	-2%
		tamanho	0.683	+5%	0.269	+14%	0.748	-3%	0.396	+9%
*Gemini-pro**	**zero-shot**	**-**	0.656	ref.	0.247	ref.	0.446	ref.	0.318	ref.
	one-shot	**aleatória**	0.574	-13%	0.254	+3%	0.655	+47%	0.366	+15%
		similaridade	0.601	-8%	0.277	+12%	0.691	+55%	0.395	+24%
		tamanho	0.601	-8%	0.305	+23%	0.609	+37%	**0.407**	**+28%**
	one-class-shot	**aleatória**	0.616	-6%	0.220	-11%	**0.848**	**+90%**	0.349	+10%
		similaridade	0.677	+3%	0.240	-3%	0.691	+55%	0.357	+12%
		tamanho	0.640	-2%	0.245	-1%	0.820	+84%	0.377	+19%
	few-shot	**aleatória**	0.697	+6%	0.285	+15%	0.633	+42%	0.393	+24%
		similaridade	**0.760**	**+16%**	**0.322**	**+30%**	0.396	-11%	0.355	+12%
		tamanho	0.690	+5%	0.296	+20%	0.583	+31%	0.392	+23%
MariTalk	**zero-shot**	**-**	**0.765**	**ref.**	0.298	ref.	0.122	ref.	0.173	ref.
	one-shot	**aleatória**	0.625	-18%	**0.320**	**+7%**	0.223	+83%	0.263	+52%
		similaridade	0.603	-21%	0.253	-15%	0.317	+160%	0.281	+62%
		tamanho	0.635	-17%	0.317	+6%	0.317	+160%	0.317	+83%
	one-class-shot	**aleatória**	0.625	-18%	0.247	-17%	0.511	+319%	**0.333**	**+92%**
		similaridade	0.536	-30%	0.201	-33%	0.446	+266%	0.277	+60%
		tamanho	0.334	-56%	0.154	-48%	0.489	+301%	0.234	+35%
	few-shot	**aleatória**	0.587	-23%	0.225	-24%	0.129	+6%	0.164	-5%
		similaridade	0.286	-63%	0.112	-62%	**0.647**	**+430%**	0.191	+10%
		tamanho	0.431	-44%	0.134	-55%	0.604	+395%	0.219	+27%

Tabela 4: Resultados dos LLMs no conjunto HateBR. Exceto pela acurácia, eles são computados na classe de discurso de ódio. Os valores em **negrito** são os melhores para cada modelo, enquanto os melhores para o conjunto estão sublinhados. As porcentagens indicam a comparação com a respectiva referência de *zero-shot*. Os resultados do Gemini* podem apresentar ligeiras flutuações devido à taxa de respostas bloqueadas pelos filtros da API do Google. Nesse conjunto, esse valor foi de 0.15%.

Modelo	#exemplos	seleção	acurácia	(±%)	precisão	(±%)	revocação	(±%)	F1	(±%)
GPT-3.5-turbo	**zero-shot**	**-**	0.528	ref.	0.452	ref.	0.679	ref.	0.542	ref.
	one-shot	**aleatória**	0.592	+12%	0.532	+18%	0.551	-19%	0.541	0%
		similaridade	**0.597**	**+13%**	0.504	+12%	0.560	-18%	0.531	-2%
		tamanho	0.586	+11%	0.495	+10%	0.642	-5%	0.559	+3%
	one-class-shot	**aleatória**	0.592	+12%	0.532	+18%	0.551	-19%	0.541	0%
		similaridade	0.553	+5%	0.504	+12%	0.640	-6%	**0.564**	**+4%**
		tamanho	0.548	+4%	0.461	+2%	**0.691**	**+2%**	0.553	+2%
	few-shot	**aleatória**	0.545	+3%	**0.599**	**+33%**	0.405	-40%	0.483	-11%
		similaridade	0.559	+6%	0.587	+30%	0.425	-37%	0.493	-9%
		tamanho	**0.597**	**+13%**	0.590	+31%	0.536	-21%	0.561	+4%
*Gemini-pro**	**zero-shot**	**-**	0.588	ref.	0.509	ref.	0.486	ref.	0.497	ref.
	one-shot	**aleatória**	0.590	0%	0.481	-6%	0.603	+24%	0.535	+8%
		similaridade	**0.607**	**+3%**	0.502	-1%	0.702	+44%	**0.554**	**+11%**
		tamanho	0.458	-22%	0.424	-17%	**0.779**	**+60%**	0.549	+10%
	one-class-shot	**aleatória**	0.545	-7%	0.494	-3%	0.525	+8%	0.509	+2%
		similaridade	0.456	-22%	0.467	-8%	0.570	+17%	0.513	+3%
		tamanho	0.478	-19%	0.442	-13%	0.671	+38%	0.533	+7%
	few-shot	**aleatória**	0.551	-6%	**0.555**	**+9%**	0.415	-15%	0.475	-4%
		similaridade	0.515	-12%	0.468	-8%	0.514	+6%	0.490	-1%
		tamanho	0.550	-6%	0.470	-8%	0.616	+27%	0.533	+7%
MariTalk	**zero-shot**	**-**	0.421	ref.	0.429	ref.	0.096	ref.	0.157	ref.
	one-shot	**aleatória**	0.434	+3%	0.591	+38%	0.128	+33%	0.211	+34%
		similaridade	0.416	-1%	**0.660**	**+54%**	0.077	-20%	0.137	-13%
		tamanho	0.429	+2%	0.602	+40%	0.131	+36%	0.215	+37%
	one-class-shot	**aleatória**	0.457	+9%	0.393	-8%	0.422	+340%	0.407	+159%
		similaridade	0.397	-6%	0.421	-2%	0.368	+283%	0.393	+150%
		tamanho	**0.482**	**+14%**	0.447	+4%	0.548	+471%	**0.492**	**+213%**
	few-shot	**aleatória**	0.381	-10%	0.587	+37%	0.274	+185%	0.374	+138%
		similaridade	0.387	-8%	0.486	+13%	0.086	-10%	0.147	-6%
		tamanho	0.478	+14%	0.441	+3%	**0.484**	**+404%**	0.462	+194%

Tabela 5: Resultados dos LLMs no conjunto OLID-BR. Exceto pela acurácia, eles são computados na classe de discurso de ódio. Os valores em **negrito** são os melhores para cada modelo, enquanto os melhores para o conjunto estão sublinhados. As porcentagens indicam a comparação com a respectiva referência de *zero-shot*. Os resultados do Gemini* podem apresentar ligeiras flutuações devido à taxa de respostas bloqueadas pelos filtros da API do Google. Nesse conjunto, esse valor foi de 0.79%.

Modelo	#exemplos	seleção	acurácia	(±%)	precisão	(±%)	revocação	(±%)	F1	(±%)
GPT-3.5-turbo	**zero-shot**	**-**	0.561	ref.	0.030	ref.	0.379	ref.	0.056	ref.
	one-shot	**aleatória**	0.555	-1%	0.029	-3%	0.500	+32%	0.055	-2%
		similaridade	0.608	+8%	0.028	-7%	0.310	-18%	0.052	-7%
		tamanho	0.610	+9%	0.042	+40%	0.293	-23%	0.074	+32%
	one-class-shot	**aleatória**	0.639	+14%	0.029	-3%	0.241	-36%	0.051	-9%
		similaridade	**0.654**	**+17%**	0.042	+40%	0.293	-23%	0.073	+30%
		tamanho	0.577	+3%	0.034	+13%	0.500	+32%	0.063	+12%
	few-shot	**aleatória**	0.647	+15%	0.017	-43%	0.121	-68%	0.029	-48%
		similaridade	0.621	+11%	**0.044**	**+47%**	0.448	+18%	**0.081**	**+45%**
		tamanho	0.572	+2%	0.040	+33%	**0.638**	**+68%**	0.075	+34%
Gemini-pro*	**zero-shot**	**-**	0.560	ref.	0.034	ref.	0.190	ref.	0.057	ref.
	one-shot	**aleatória**	0.457	-18%	0.045	+32%	0.310	+63%	0.078	+37%
		similaridade	0.619	+11%	0.040	+18%	0.421	+122%	0.073	+28%
		tamanho	0.636	+14%	0.046	+35%	0.414	+118%	0.082	+44%
	one-class-shot	**aleatória**	0.584	+4%	0.052	+53%	0.397	+109%	0.092	+61%
		similaridade	0.588	+5%	0.039	+15%	**0.586**	**+208%**	0.074	+30%
		tamanho	**0.676**	**+21%**	0.057	+68%	0.276	+45%	0.095	+67%
	few-shot	**aleatória**	0.609	+9%	**0.059**	**+74%**	0.310	+63%	**0.100**	**+75%**
		similaridade	0.641	+14%	0.048	+41%	0.466	+145%	0.086	+51%
		tamanho	0.630	+12%	0.058	+71%	0.333	+75%	0.099	+74%
MariTalk	**zero-shot**	**-**	**0.501**	ref.	0.028	ref.	0.052	ref.	0.036	ref.
	one-shot	**aleatória**	0.493	-2%	0.050	+79%	0.034	-35%	0.041	+14%
		similaridade	0.496	-1%	**0.087**	**+211%**	0.069	+33%	**0.077**	**+114%**
		tamanho	0.492	-2%	0.055	+96%	0.052	+0%	0.053	+47%
	one-class-shot	**aleatória**	0.474	-5%	0.026	-7%	0.121	+133%	0.043	+19%
		similaridade	0.495	-1%	0.043	+54%	0.224	+331%	0.072	+100%
		tamanho	0.316	-37%	0.012	-57%	0.397	+663%	0.025	-31%
	few-shot	**aleatória**	0.447	-11%	0.014	-50%	0.121	+133%	0.025	-31%
		similaridade	0.223	-55%	0.037	+32%	0.069	+33%	0.048	+33%
		tamanho	0.029	-94%	0.014	-50%	**0.931**	**+1690%**	0.027	-25%

Tabela 6: Resultados dos LLMs no conjunto ToLD-Br. Exceto pela acurácia, eles são computados na classe de discurso de ódio. Os valores em **negrito** são os melhores para cada modelo, enquanto os melhores para o conjunto estão sublinhados. As porcentagens indicam a comparação com a respectiva referência de *zero-shot*. Os resultados do Gemini* podem apresentar ligeiras flutuações devido à taxa de respostas bloqueadas pelos filtros da API do Google. Nesse conjunto, esse valor foi de 0.12%.

se responder à terceira questão de pesquisa elaborada — *Adicionar contexto extra e demonstrações selecionadas de maneira acurada aprimora a resposta de grandes modelos de linguagem generativos acessados via prompt?*.

Especialmente, um aspecto do modelo Gemini-pro demanda atenção. Durante a realização de inferências por meio da API, verificou-se que algumas solicitações foram bloqueadas devido à política de filtros a conteúdos sensíveis do Google, mesmo quando as configurações de bloqueio estavam minimizadas. Inicialmente, poder-se-ia presumir que a instância em avaliação contivesse ao menos um elemento ofensivo, dada essa ocorrência; no entanto, essa suposição não é necessariamente acurada. O bloqueio pode ter sido ocasionado pelo conjunto de exemplos fornecidos no *prompt*. Contudo, visto que as taxas de bloqueio foram extremamente baixas, optou-se por manter os cálculos das métricas apenas desconsiderando as instâncias bloqueadas. Ainda que isso cause uma flutuação mínima nos resultados métricos, uma comparação com os outros modelos não deve ser relevantemente prejudicada.

Outrossim, mesmo que o MariTalk seja fundamentado em um modelo de linguagem pré-treinado em português, os modelos GPT-3.5-turbo e Gemini-pro demonstram desempenhos superiores na maioria dos casos. Entretanto, os detalhes específicos acerca dos conjuntos de dados utilizados no pré-treinamento desses dois últimos modelos não são publicamente disponíveis, o que representa uma limitação significativa para a compreensão plena das origens desse fato.

Além disso, em um cenário em que o desbalanceamento de classes constitui uma preocupação e a minimização tanto de falsos positivos quanto de falsos negativos é imperativa, a métrica F1, particularmente para a classe de discurso de ódio, requer um enfoque particular. Para essa métrica, nos conjuntos de dados HateBR e OLID-BR, tanto o GPT-3.5-turbo quanto o MariTalk alcançaram desempenhos superiores na configuração de *one-class-shot*, enquanto o Gemini-pro exibiu melhores resultados na configuração de *one-shot*. No conjunto ToLD-Br, os modelos GPT-3.5-turbo e Gemini-pro alcançaram os seus melhores resultados utilizando a abordagem *few-shot*, ao passo que o MariTalk sobressaiu-se na abordagem *one-shot*. Essa variabilidade pode ser atribuída às diferenças na capacidade de cada modelo de integrar e responder ao número de informações fornecidas via *prompt*.

Dados	Modelo	acurácia	(±%)	precisão	(±%)	revocação	(±%)	F1	(±%)
HateBR	**GPT-3.5 one-class-shot/tam.**	0.697	-	0.278	-	**0.763**	-	0.408	-
	GPT-3.5 one-class-shot/tam. + ctx.	**0.765**	**+10%**	**0.328**	**+18%**	0.633	-17%	**0.432**	**+6%**
	Gemini-pro one-shot/tam.	0.601	-	0.305	-	**0.609**	-	0.407	-
	Gemini-pro one-shot/tam. + ctx.	**0.783**	**+30%**	**0.393**	**+29%**	0.424	-30%	**0.408**	**0%**
	MariTalk one-class-shot/aleat.	**0.625**	-	**0.247**	-	0.511	-	**0.333**	-
	MariTalk one-class-shot/aleat. + ctx.	0.454	-27%	0.152	-38%	**0.748**	**+46%**	0.253	-24%
OLID-BR	**GPT-3.5 one-class-shot/sim.**	0.533	-	0.504	-	**0.640**	-	**0.564**	-
	GPT-3.5 one-class-shot/sim. + ctx.	**0.626**	**+17%**	**0.560**	**+11%**	0.563	-12%	0.562	0%
	Gemini-pro one-shot/sim.	**0.607**	-	0.502	-	**0.702**	-	**0.554**	-
	Gemini-pro one-shot/sim. + ctx.	0.605	0%	**0.600**	**+20%**	0.267	-62%	0.369	-33%
	MariTalk one-class-shot/tam.	**0.482**	-	**0.447**	-	0.548	-	**0.492**	-
	MariTalk one-class-shot/tam. + ctx.	0.312	-35%	0.317	-29%	**0.884**	**+61%**	0.466	-5%
ToLD-Br	**GPT-3.5 few-shot/sim.**	0.621	-	0.044	-	0.448	-	0.081	-
	GPT-3.5 few-shot/sim. + ctx.	**0.671**	**+8%**	**0.070**	**+59%**	**0.483**	**+8%**	**0.123**	**+52%**
	Gemini-pro few-shot/aleat.	0.609	-	**0.059**	-	**0.310**	-	**0.100**	-
	Gemini-pro few-shot/aleat. + ctx.	**0.680**	**+12%**	0.042	-29%	0.190	-39%	0.069	-31%
	MariTalk one-shot/sim.	**0.496**	-	**0.087**	-	0.069	-	**0.077**	-
	MariTalk one-shot/sim. + ctx.	0.065	-87%	0.015	-83%	**0.966**	**+1.3k%**	0.029	-62%

Tabela 7: Resultados da adição de contexto extra às melhores configurações dos LLMs ativados via *prompt* com base no F1. Exceto pela acurácia, eles são computados na classe de discurso de ódio. Os valores em **negrito** representam os melhores resultados quanto à estratégia de adição de contexto, enquanto os melhores para todo o conjunto de dados estão sublinhados. O valor de porcentagem indica a comparação com a respectiva referência, nesse caso, a configuração sem contexto adicional.

Por exemplo, no conjunto ToLD-Br, dois dos modelos exigiram um número maior de exemplos, constatado pelo melhor desempenho ao empregar a estratégia *few-shot*. Isso se dá em um contexto no qual, novamente, os textos provêm exclusivamente de uma rede cujos usuários se expressam primordialmente de maneira textual, o que pode implicar a presença de uma maior diversidade de recursos linguísticos. Em contraste, para os conjuntos de dados HateBR e OLID-BR, nenhum dos melhores desempenhos dos modelos foi nessa configuração.

No que concerne à seleção de demonstrações para os *prompts*, de modo geral, todos os modelos beneficiaram-se de estratégias baseadas na similaridade e tamanho dos exemplos, com apenas dois casos em que a seleção aleatória proporcionou os melhores resultados para a medida F1. Essa constatação reitera que as estratégias de seleção de elementos que compõem o *prompt* impactam na otimização do desempenho dos modelos de linguagem em tarefas específicas. Desse modo, também foram acrescentadas informações contextuais aos melhores resultados das estratégias de seleção de demonstrações. Isso é feito para verificar se é possível aprimorar ainda mais a capacidade de resposta dos modelos ao fornecer palavras-chave adicionais em conjunto dos exemplos demonstrativos. A Tabela 7 sintetiza esses resultados.

Em termos gerais, o modelo GPT-3.5-turbo, que foi pré-treinado predominantemente em inglês, apresentou uma melhoria significativa com a incorporação de contexto adicional em seus *prompts*. O modelo estabeleceu, inclusive, novos melhores resultados para a métrica F1 nos conjuntos de dados HateBR e ToLD-Br. Por sua vez, o Gemini-pro obteve um aumento na acurácia e na precisão no conjunto HateBR, melhorias em precisão para o OLID-BR, e um incremento na acurácia para o ToLD-BR. Já o MariTalk exibiu maiores resultados sobre a revocação em todos os conjuntos. Assim, a utilização de contextos adicionais pode constituir uma estratégia complementar para mitigar a incidência de falsos positivos e falsos negativos, dependendo do contexto específico almejado.

5.4. Resultados Comparativos por Classe

Esta seção apresenta uma análise comparativa centrada nos resultados da métrica F1 para os melhores representantes de cada grupo de modelos: um representante dos classificadores ba-

Classes →	**Neutro**		**Ofensivo**		**Discurso de Ódio**	
Dados ↓	**Modelo**	**F1**	**Modelo**	**F1**	**Modelo**	**F1**
HateBR	Bernice *(ajuste fino)*	**0.906**	BERTimbau *(ajuste fino)*	**0.861**	BERTimbau *(ajuste fino)*	**0.667**
	MariTalk *(one-shot/sim.)*	0.899	MariTalk *(one-shot/sim.)*	0.793	GPT-3.5-turbo *(one-class-shot/tam. + ctx.)*	0.432
	Gervásio-7B-PTBR *(ajuste fino)*	0.770	Gervásio-7B-PTBR *(ajuste fino)*	0.672	Gervásio-7B-PTBR *(ajuste fino)*	0.345
OLID-BR	BERTweet.BR *(ajuste fino)*	0.547	Bernice *(ajuste fino)*	**0.765**	BERTweet.BR *(ajuste fino)*	**0.606**
	MariTalk *(one-shot/sim.)*	**0.605**	GPT-3.5-turbo *(one-shot/sim.)*	0.719	GPT-3.5-turbo *(one-class-shot/sim.)*	0.564
	Gervásio-7B-PTBR *(ajuste fino)*	0.304	Sabiá-1-7B *(ajuste fino)*	0.657	Gervásio-7B-PTBR *(ajuste fino)*	0.475
ToLD-Br	Bernice *(ajuste fino)*	0.747	BERTweet.BR *(ajuste fino)*	**0.731**	BERTweet.BR *(ajuste fino)*	**0.178**
	MariTalk *(few-shot/sim.)*	**0.798**	MariTalk *(one-shot/tam.)*	0.701	GPT-3.5-turbo *(few-shot/sim. + ctx.)*	0.123
	Gervásio-7B-PTBR *(ajuste fino)*	0.504	Sabiá-1-7B *(ajuste fino)*	0.572	Gervásio-7B-PTBR *(ajuste fino)*	0.041

Tabela 8: Resultados dos melhores modelos para cada classe com base na métrica de F1. Foi selecionado um modelo para cada grupo avaliado, *i.e.*, um representante dos classificadores baseados em *encoder*, um dos classificadores baseados em *decoder* e um dos LLMs ativados via *prompts*. Os valores em **negrito** são os melhores para cada conjunto.

seados em *encoder*, um dos classificadores baseados em *decoder* e um dos LLMs ativados via *prompts*. São reportados valores para as classes neutro, ofensivo e discurso de ódio. As análises anteriores não incluíam as outras classes, principalmente, devido à relevância e aos desafios intrínsecos associados à categoria de discurso de ódio neste contexto, haja vista também o entrave do desbalanceamento de dados, o que dificulta a apresentação de uma média geral. Os resultados detalhados são apresentados na Tabela 8.

Entre os nove casos analisados, o modelo BERTweet.BR se destacou em três, enquanto Bernice e BERTimbau lideraram em dois casos cada. Notavelmente, na classe de discursos de ódio, o modelo pertencente à família GPT apresentou os melhores desempenhos para seu grupo. Por outro lado, nas categorias que não o discurso de ódio, o MariTalk prevaleceu como o modelo representante dos LLMs ativados via *prompt*, com exceção da classe ofensivo para o OLID-BR. Isso pode indicar que, devido ao seu *corpus* de pré-treinamento em português, o MariTalk possua uma melhor compreensão de conteúdos cotidianos não odiosos, enquanto o modelo GPT, com sua reconhecida capacidade de entendimento contextual, possa discernir de forma mais eficaz as nuances do discurso de ódio. É importante salientar que o Gemini-pro não se destacou em nenhum dos casos analisados. Quanto aos classificadores baseados em *decoder*, não foi observada alguma situação em que esse grupo superasse os demais. Tal ausência sugere que a abordagem adotada, que considera sua capacidade de alinhamento semântico, pode ser insuficiente para suas aplicações como classificadores convencionais, exigindo alterações arquiteturais conforme observado em trabalhos mais recentes (BehnamGhader et al., 2024). Porém, destaca-se que o Gervásio-7B superou o Sabiá-1-7B, que liderou apenas em dois dos nove casos dentre os classificadores baseados em *decoder*.

Os resultados corroboram a maior eficácia dos modelos baseados em *encoder* aliados à estratégia de ajuste fino. No entanto, a Tabela 9 evidencia que, mesmo entre os modelos de melhor desempenho, ainda há ocorrência de confusão entre as classes analisadas. No caso do conjunto HateBR, observa-se uma maior taxa de predições corretas, embora a principal dificuldade resida na classificação da classe "discurso de ódio". Nesse contexto, os erros de predição encontram-se equilibrados entre as classes "neutro" e "ofensivo". Em relação ao conjunto OLID-BR, a dificuldade do modelo torna-se mais evidente nas classes limítrofes, consideradas em termos de severidade, especificamente de "neutro" para "ofensivo", assim como entre "ofensivo" e "discurso de ódio". Esse cenário destaca a dificuldade do modelo em estabelecer limites claros entre essas classes. Já no conjunto ToLD-BR, verifica-se uma dificuldade significativa na separação da classe majoritária ("neutro") em relação às demais.

(a) BERTimbau – HateBR

Categoria Verdadeira	Categoria Predita		
	Neutro	Ofensivo	Ódio
Neutro	601	52	29
Ofensivo	37	469	28
Ódio	7	34	98

(b) BERTweet.BR – OLID-BR

Categoria Verdadeira	Categoria Predita		
	Neutro	Ofensivo	Ódio
Neutro	143	38	23
Ofensivo	107	550	120
Ódio	69	98	238

(c) BERTweet.BR – ToLD-BR

Categoria Verdadeira	Categoria Predita		
	Neutro	Ofensivo	Ódio
Neutro	1513	604	210
Ofensivo	265	1295	70
Ódio	10	15	33

Tabela 9: Matrizes de confusão dos melhores modelos *encoder*, selecionados com base na frequência de maiores valores de F1 entre as classes (Tabela 8), para cada conjunto de dados.

Para mais, é proeminente a discrepância nos valores observados para a categoria de discurso de ódio no conjunto de dados ToLD-Br em comparação com os conjuntos HateBR e OLID-BR. Essa diferença pode estar associada, para além do desequilíbrio de classes, à origem dos dados. Como supracitado, o ToLD-Br provém exclusivamente do Twitter, ao passo que o OLID-BR inclui também outra fonte, o YouTube, e o HateBR contempla comentários do Instagram. Todavia, essa variação também pode ser atribuída ao processo de construção dos conjuntos e rotulação dos dados, influenciada, entre outros aspectos, pela subjetividade dos anotadores. Para investigar esse fenômeno, além de elucidar ainda mais o comportamento dos modelos, a próxima seção contempla uma inspeção qualitativa de algumas instâncias classificadas.

5.5. Inspeção Qualitativa

A Tabela 10 apresenta instâncias selecionadas dos conjuntos de dados, incluindo seus respectivos rótulos e as predições realizadas pelos modelos, especificamente aqueles que obtiveram os melhores resultados para a métrica F1 para a classe discurso de ódio, consoante exibido na Tabela 8. O propósito dessa análise é proporcionar uma perspectiva qualitativa sobre como os modelos se comportam em comparação com a rotulação realizada por humanos. ⚠ **É importante salientar novamente a possibilidade de presença de conteúdo ofensivo e potencialmente nocivo nas instâncias examinadas.**

Um registro específico do conjunto HateBR destaca-se ao exibir o uso do termo "ordinária". Embora em seu sentido literal o termo denote algo "habitual, comum e corriqueiro", em um contexto figurado pode significar "mau caráter"[24]. Na rotulação original, esse caso foi identificado como discurso de ódio, possivelmente por acompanhar a palavra "mulher". No entanto, os modelos atribuíram a essa ocorrência a classificação de ofensivo. Essa decisão dos modelos parece plausível, visto que, mesmo com a inclusão de "mulher", não há evidência explícita de discriminação baseada em gênero, o que caracterizaria sexismo[25] e, por extensão, discurso de ódio. De forma semelhante, outros registros do HateBR foram classificadas como discurso de ódio pelos rotuladores, possivelmente devido à presença de expressões como "gay enrustido", enquanto os modelos as categorizaram como ofensivas apenas.

Outro caso intrigante envolve uma referência à ex-presidente do Chile, Michelle Bachelet, utilizando o termo "senhora". Embora essa instância tenha sido rotulada como ofensiva, potencialmente sugerindo um viés de etarismo, os modelos a classificaram como neutra. Esta divergência é particularmente controversa, uma vez que o termo em questão pode ser empregado simplesmente como uma forma de tratamento respeitoso. Nesse contexto, os modelos podem não capturar a sutileza e a gravidade atribuída ao uso de certos termos da mesma maneira que um anotador humano. Não obstante, esse exemplo também pode mostrar um caso de subjetividade particular entre indivíduos.

Adicionalmente, duas sentenças que mencionam religião em um teor negativo foram classificadas como discurso de ódio pelos modelos, enquanto originalmente uma é considerada neutra no HateBR e a outra ofensiva no OLID-BR. Esse cenário destaca a necessidade de investigações sobre vieses sociais relacionados à religião, tanto em conjuntos de dados quanto eventualmente nos modelos preditivos. Além disso, múltiplos casos, sobretudo no ToLD-Br, empregam o termo "nego(a)" para referir-se a

[24] Significado de ordinário: https://www.significados.com.br/ordinario/

[25] Definição de sexismo: https://www.significados.com.br/sexismo/

Dados	Texto	Rótulo	Predição
HateBR	Q mulher ordinária	discurso de ódio	ofensivo
	esses caras dão muito é CÚ	discurso de ódio	ofensivo
	Bandido gay enrustido ! Pedófilo de merda !	discurso de ódio	ofensivo
	Senhora Michele Bachelet !	ofensivo	neutro
	Quando alguém fala que é cristão , crispa até o cabelo da orelha .	neutro	discurso de ódio
OLID-BR	KKKKKKKKKK falo tudo cara ... a mulher tenta fuder toda a porra da história dos lobisomens e vampiros !	discurso de ódio	ofensivo
	O nega besta mds	discurso de ódio	ofensivo
	chorando igual uma vagabunda porra eu sabia q ia ser triste mas nao esperava essa cenakkkkkkkk	discurso de ódio	ofensivo
	Cada dia gosto respeito menos as religiões , todas . Bando de cínicos aproveitadores . URL	ofensivo	discurso de ódio
	USER Tinham sim , nos campos de concentração os gays usavam um triângulo rosa na roupa inclusive .	neutro	discurso de ódio
ToLD-Br	eu tô só adiando pra ir na casa de vovó , a bichinha já nem acredita quando falo que vou lá	discurso de ódio	neutro
	se ngm for cmg nessa feirinha vou ficar mt puta pq todo dia nego me cobra pra ir lá	discurso de ódio	neutro
	@USER Sim , tlg que foi uma brincadeira ... Mas eu imagino se fosse ao contrário , slc , nego n ia querer saber se o vídeo tá cortado ou não , já ia falar q o ney eh lixo	discurso de ódio	ofensivo
	alguém , além de mim , consegue morder a língua dormindo ? ? ? pqp !	ofensivo	neutro
	pai comprou bud e pqp q coisa boa	ofensivo	neutro

Tabela 10: Exemplos de predições em que os modelos — os mais eficazes entre os classificadores baseados em *encoder*, *decoder* e LLM ativado via *prompt* — concordam entre si nas predições, mas divergem do rótulo verdadeiro. A escolha dos modelos foi baseada na métrica F1 para a classe de discurso de ódio em cada conjunto de dados.

alguém, e foram identificados pelos anotadores humanos como discurso de ódio. Apesar de o termo não ser necessariamente usado para expressar racismo, alguns estudos sugerem que seu uso deveria ser evitado devido a sua conotação histórica (Guimarães Nascimento & Ribeiro, 2018). As instâncias que utilizaram esse termo foram percebidas como ofensivas ou neutras pelos modelos.

Uma instância específica do OLID-BR suscita preocupações. Embora trate de um assunto altamente sensível, referindo-se às vestimentas usadas por homossexuais em campos de concentração nazistas, seu conteúdo é, aparentemente, apresentado de forma explicativa e factual. Originalmente, essa instância recebeu o rótulo neutro, mas os modelos a classificaram como discurso de ódio. Esse é um caso que poderia acarretar uma possível censura equivocada devido à falha dos modelos em discernir o contexto adequadamente. Por outro lado, outro exemplo do OLID-BR apresenta o emissor se referindo a sua avó com o termo "bichinha". Como discutido anteriormente, embora esse termo possa ser associado a discursos homofóbicos, ele também é empregado com uma conotação carinhosa em certas regiões[26], o que parece ser o caso. Essas ocorrências demonstram que a tarefa de interpretar corretamente o contexto e a intenção imputada no uso de certos termos é complexa não só para os modelos, como também para os humanos.

Ademais, frequentemente sentenças que contêm a expressão "pqp" têm seus rótulos oficiais como ofensivo, porém são atribuídas para a classe neutro pelos modelos. Essa é uma expressão considerada vulgar que, embora possa ser utilizada como interjeição, pode ser ofensiva para determinados públicos. Neste contexto, é possível que os modelos tenham falhado em captar nuances semânticas associadas ao termo, com obstáculos associados ao seu pequeno

[26]Uso do termo "bichinho": https://pt.m.wiktionary.org/wiki/bichim

comprimento —apenas três letras— e ao fato de ser uma sigla cujo significado completo não é claramente manifestado[27].

Assim, os exemplos analisados ilustram a complexidade desta tarefa, que envolve o reconhecimento e a inferência adequada de uma variedade de aspectos culturais, sociais e históricos. Eles também revelam que os modelos podem não captar nuances extremamente específicas e indicam que existem desafios até mesmo na elaboração dos conjuntos de dados.

6. Conclusão

Este trabalho investigou o uso de classificadores fundamentados em *encoders* e *decoders*, especialmente ajustados para identificar discursos de ódio em português do Brasil, além de explorar diversas adaptações de *prompts* para ativar LLMs. Visando enriquecer a semântica na ativação de modelos generativos, propomos duas abordagens para o aprimoramento dos *prompts*: a incorporação de palavras-chave por meio de modelagem de tópicos e a seleção de exemplos demonstrativos de forma criteriosa. Observamos que, apesar da recente popularidade dos LLMs e suas proclamadas habilidades contextuais que supostamente dispensam treinamento extensivo, a adaptação de classificadores baseados em *encoder* de menor escala alcança os melhores resultados, especialmente com a estratégia de ajuste fino. Avaliamos também a eficácia de modelos baseados em *decoder* como classificadores tradicionais, considerando suas capacidades de alinhamento semântico. Constatamos que essa abordagem produziu os piores resultados no geral, indicando que ela pode não ser suficiente para contextos que exigem a captura de nuances delicadas. Além disso, as estratégias de escolha de exemplos demonstrativos e a adição de contexto beneficiam os modelos generativos ativados via *prompt*, com as abordagens sem demonstrações ou com escolhas aleatórias sendo superadas na maioria dos casos. Por fim, a detecção de discurso de ódio utilizando *transformers* mostra-se promissora, mas ainda enfrenta desafios, evidentes sobretudo na inspeção qualitativa realizada.

Em vista do exposto, reforçamos a literatura recente que defende uma investigação aprofundada sobre a capacidade dos modelos de linguagem de lidar com padrões sociais delicados, como o discurso de ódio, especialmente no idioma português. Modelos menores ainda têm um papel importante para evitar a perpetuação de questões sociais em ferramentas de PLN. Estudos futuros poderiam aprofundar-se em analisar o impacto das camadas, do *corpus* de treinamento e das variações arquiteturais nos modelos BERTimbau, BERTweet.BR, AlBERTina e Bernice, buscando elucidar as razões pelas quais esses modelos conseguem processar de maneira mais eficaz esse tipo de conteúdo. O desempenho variável de diferentes modelos em distintos conjuntos de dados, aliado à divergência observada nos resultados entre conjuntos de origens variadas, destaca a importância de se aprofundar no estudo da variabilidade de comportamentos e estratégias nas diversas redes sociais. Ademais, embora se tenha avaliado muitas abordagens, novos trabalhos podem incluir estratégias que considerem comitês e fusão de modelos, assim como a combinação de conjuntos de dados para treinamento (Leonardelli et al., 2023). Por fim, trabalhos subsequentes também incluem a investigação de novas adaptações arquiteturais que aprimoram a habilidade de modelos baseados em *decoder* em atuar como classificadores, bem como das nossas propostas para estratégias de aperfeiçoamento de *prompts*, visando verificar sua aplicabilidade em outros contextos.

6.1. Limitações

Este estudo possui limitações quanto à divisão de seus conjuntos de treinamento, validação e teste, pois adota apenas uma partição única. Essa restrição decorre principalmente dos custos associados ao uso da API do GPT, da API Gemini e dos recursos computacionais no ambiente Google Cloud. Ademais, essa limitação também restringiu a variação dos hiperparâmetros para nossos modelos, como o ajuste do número de épocas ou a modificação dos parâmetros de aprendizagem e inferência. Estamos cientes de que esses fatores podem influenciar direta ou indiretamente a interpretação do comportamento dos modelos em cenários mais abrangentes e generalistas. Todavia, tais decisões viabilizaram a análise e comparação de diversas abordagens, cada uma com suas características particulares. Outro aspecto relevante é que LLMs, como os avaliados neste trabalho, utilizam *corpora* amplamente baseados em dados extraídos da *web* em sua etapa de pré-treinamento. Ainda que não tenhamos analisado a contaminação potencial desses dados em relação aos conjuntos experimentados, os resultados indicam que esses modelos não incorporaram os rótulos para os textos, dado o desempenho registrado. Dessa forma, contribuímos com uma análise comparativa entre uma vasta quantidade de modelos sobre uma tarefa de classificação extremamente sensível.

[27] Uso do termo "pqp": https://pt.wiktionary.org/wiki/puta_que_pariu

Declaração Ética

Tratando-se de conjuntos de dados provenientes de redes sociais, procedeu-se à anonimização adequada de usuários e *links*. Além disso, considerando a sensibilidade associada ao tema de discurso de ódio, todas as pessoas que, de alguma forma, acessaram os dados foram previamente cientificadas sobre a sua natureza delicada. Para mais, o domínio exige considerações criteriosas. Primeiro, conjuntos de dados podem abrigar vieses culturais e históricos, falhando em capturar a plenitude da diversidade linguística e cultural. Nesse aspecto, o Brasil é um exímio exemplo de diversidade cultural; meramente examinar diferentes perspectivas dentro de regiões do mesmo país pode revelar discrepâncias na ofensividade percebida de um termo. Ademais, ao considerar-se perspectivas inter-países, tais diferenças podem se tornar ainda mais acentuadas, mesmo entre falantes do mesmo idioma. Por exemplo, "rapariga" em Portugal significa primariamente "jovem mulher", enquanto, no Brasil, o termo pode carregar conotações depreciativas[28].

Além disso, a classificação incorreta de conteúdos ofensivos e de discurso de ódio podem acarretar implicações significativas — falsos positivos podem levar a censura indevida e falsos negativos podem falhar em proteger grupos vulneráveis. Ao se classificar um conteúdo neutro como discurso de ódio pode-se incorrer em impactos para a sociedade, tais como, o desencorajamento dos usuários de se expressarem livremente, temendo a censura e a perda de confiança nas plataformas como espaço de debate público. Muitas vezes as redes sociais são a única ferramenta que as minorias encontram para lutar contra as opressões sofridas. Ao suprimir discussões sobre tópicos sensíveis, pode-se sufocar este debate público e impedir o avanço social em pautas importantes para estes grupos, tais como racismo, sexismo e violências enfrentadas em seu meio. Socialmente o impacto deste tipo de falso positivo pode aumentar desigualdades, suprimir o direito à liberdade e ao espaço democrático, com tendência de afetar desproporcionalmente os grupos sub-representados. Por outro lado, ao se classificar o discurso de ódio como neutro, pode-se perpetuar a atuação de grupos extremistas, a incitação à violência e o assédio, amplificando o impacto social e o sofrimento psicológico dos indivíduos. Neste caso, novamente, as comunidades vulneráveis sofreriam com maiores consequências.

Por fim, como forma de minimizar as preocupações éticas sobre falsos positivos e negativos, enfatizamos que mecanismos de Inteligência Artificial (IA) são viáveis enquanto auxílios na moderação de conteúdo, mas não devem ser substitutos definitivos para ela. Destaca-se a necessidade da observância dos princípios éticos, com destaque para o desenvolvimento de abordagens explicáveis e transparentes. Modelos que fornecem explicações para suas decisões podem ajudar os moderadores a entender por que o conteúdo foi sinalizado e fazer julgamentos mais bem informados. Além disso, os usuários das plataformas precisam estar cientes de que seu conteúdo foi moderado com o auxílio de IA. Alcançar o equilíbrio certo é complexo, exigindo ainda avanços na técnica e robustez dos modelos e na supervisão ética de equipes interdisciplinares.

Agradecimentos

Os autores agradecem o financiamento do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), bolsas 307088/2023-5 e 315750/2021-9, da FAPERJ - Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro, processos SEI-260003/002930/2024, SEI-260003/000614/2023, SEI-260003/006057/2024, e da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código Financeiro 001. Também contou-se com o apoio do programa *Google Cloud Research Credits*, sob o código GCP19980904, e do programa de créditos da Maritaca AI. Por fim, utilizamos o ChatGPT-4 para agilizar a elaboração de códigos em Python e Latex. Cada sugestão da IA foi cuidadosamente examinada, testada e, muitas vezes, ajustada por nós, imputando-nos a plena responsabilidade pela forma e conteúdo deste artigo.

Os autores agradecem ainda aos revisores e ao editor pelas observações e sugestões construtivas, que ajudaram a melhorar o artigo.

Referências

Almeida, Thales Sales, Hugo Abonizio, Rodrigo Nogueira & Ramon Pires. 2024. Sabiá-2: A new generation of portuguese large language models. arXiv [cs.CL]. 10.48550/arXiv.2403.0988

Aluru, Sai Saketh, Binny Mathew, Punyajoy Saha & Animesh Mukherjee. 2020. Deep learning models for multilingual hate speech detection. arXiv [cs.SL/cs.CL]. 10.48550/arXiv.2004.06465

[28] https://bit.ly/rapariga-Brasil-Portugal

Amorim, Annie, Nils Murrugarra-Llerena, Vítor Silva, Daniel de Oliveira & Aline Paes. 2023. ALTES: uma ferramenta de rotulação automática de tópicos por meio de fontes externas. Em *38th Simpósio Brasileiro de Bancos de Dados*, 120–125. 10.5753/sbbd_estendido.2023.233252

Assis, Gabriel, Annie Amorim, Jonnathan Carvalho, Daniel de Oliveira, Daniela Vianna & Aline Paes. 2024. Exploring Portuguese hate speech detection in low-resource settings: Lightly tuning encoder models or in-context learning of large models? Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 301–311.

BehnamGhader, Parishad, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados & Siva Reddy. 2024. LLM2Vec: Large language models are secretly powerful text encoders. arXiv [cs.CL]. 10.48550/arXiv.2404.05961

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever & Dario Amodei. 2020. Language models are few-shot learners. Em *Neural Information Processing Systems*, 1877–1901.

Caneiro, Fernando, Daniela Viana, Jonathan Carvalho, Alexandre Plastino & Aline Paes. 2024. BERTweet.BR: A pretrained language model for tweets in Portuguese. *Neural Computing and Applications*. 10.1007/s00521-024-10711-3

Chiu, Ke-Li, Annie Collins & Rohan Alexander. 2022. Detecting hate speech with GPT-3. arXiv [cs.CL]. 10.48550/arXiv.2103.12407

Das, Mithun, Saurabh Kumar Pandey & Animesh Mukherjee. 2023. Evaluating ChatGPT's performance for multilingual and emoji-based hate speech detection. arXiv [cs.CL]. 10.48550/arXiv.2305.13276

DeLucia, Alexandra, Shijie Wu, Aaron Mueller, Carlos Aguirre, Philip Resnik & Mark Dredze. 2022. Bernice: A multilingual pre-trained encoder for twitter. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6191–6205. 10.18653/v1/2022.emnlp-main.415

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Em *Conference of the North American Chapter of the Association for Computational Linguistics (AACL)*, 4171–4186. 10.18653/v1/N19-1423

Edwards, Griffin Sims & Stephen Rushin. 2018. The effect of president Trump's election on hate crimes. *SSRN Electronic Journal* 10.2139/ssrn.3102652

Fortuna, Paula & Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys* 51(4). 1–30. 10.1145/3232676

Fortuna, Paula, João Rocha da Silva, Juan Soler-Company, Leo Wanner & Sérgio Nunes. 2019. A hierarchically-labeled Portuguese hate speech dataset. Em *3rd Workshop on Abusive Language Online*, 94–104. 10.18653/v1/W19-3510

de Freitas, Riva Sobrado & Matheus Felipe de Castro. 2013. Liberdade de expressão e discurso do Ódio: um exame sobre as possíveis limitações à liberdade de expressão. *Seqüência Estudos Jurídicos e Políticos* 34(66). 327–355. 10.5007/2177-7055.2013v34n66p327

French, Megan & Natalya N. Bazarova. 2017. Is Anybody out There?: Understanding Masspersonal Communication through Expectations for Response across Social Media Platforms. *Journal of Computer-Mediated Communication* 22(6). 303–319. 10.1111/jcc4.12197

Fu, Jinlan, See-Kiong Ng, Zhengbao Jiang & Pengfei Liu. 2024. GPTScore: Evaluate as you desire. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 6556–6576. 10.18653/v1/2024.naacl-long.365

Google, Gemini Team. 2023. Gemini: A family of highly capable multimodal models. arXiv [cs.CL]. 10.48550/arXiv.2312.11805

Grootendorst, Maarten. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv [cs.CL]. 10.48550/arXiv.2203.05794

Guimarães Nascimento, Raquel Costa & Erislane Rodrigues Ribeiro. 2018. Uma análise discursiva dos memes "nego isso, nego aquilo". *Revista do Sell* 7(1).

Hu, Edward J, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang & Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. Em *International Conference on Learning Representations*,

Jahan, Md Saroar & Mourad Oussalah. 2023. A systematic review of hate speech automatic detection using natural language processing. *Neurocomputing* 546. 126232. 10.1016/j.neucom.2023.126232

Jin, Xin & Jiawei Han. 2011. K-Means clustering. *Encyclopedia of Machine Learning* 563–564. 10.1007/978-0-387-30164-8_425

Leite, João Augusto, Diego Silva, Kalina Bontcheva & Carolina Scarton. 2020. Toxic Language Detection in Social Media for Brazilian Portuguese: New Dataset and Multilingual Analysis. Em *1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, 914–924. 10.18653/v1/2020.aacl-main.91

Leonardelli, Elisa, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, Alexandra Uma & Massimo Poesio. 2023. SemEval-2023 task 11: Learning with disagreements (LeWiDi). Em *17th International Workshop on Semantic Evaluation (SemEval)*, 2304–2318. 10.18653/v1/2023.semeval-1.314

Li, Zongxi, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu Lee Wang, Qing Li & Xiaoqin Zhong. 2023. Label supervised LLaMA finetuning. arXiv [cs.CL]. 10.48550/arXiv.2310.01208

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi & Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys* 55(9). 1–35. 10.1145/3560815

Moura, Marco Aurelio. 2016. *O discurso do ódio em redes sociais*. Lura Editorial

Nguyen, Thanh Thi, Campbell Wilson & Janis Dalins. 2023. Fine-tuning Llama 2 large language models for detecting online sexual predatory chats and abusive texts. arXiv [cs.CL]. 10.48550/arXiv.2308.14683

Oliveira, Amanda, Thiago de Carvalho Cecote, João Paulo Reis Alvarenga, Vander Luis de Souza Freitas & Eduardo José da Silva Luz. 2024. Toxic speech detection in Portuguese: A comparative study of large language models. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 108–116.

Oliveira, Amanda, Thiago Cecote, Pedro Silva, Jadson Gertrudes, Vander Freitas & Eduardo Luz. 2023. How good is ChatGPT for detecting hate speech in Portuguese? Em *14th Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, 94–103. 10.5753/stil.2023.233943

Ouyang, Long, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike & Ryan Lowe. 2022. Training language models to follow instructions with human feedback. Em *36th Conference on Neural Information Processing Systems (NeurIPS)*, 27730–27744.

Paiva, Peter, Vanecy da Silva & Raimundo Moura. 2019. Detecção automática de discurso de ódio em comentários online. Em *7th Escola Regional de Computação Aplicada à Saúde*, 157–162.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot & E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12. 2825–2830.

Pelle, Rogers, Cleber Alcântara & Viviane P. Moreira. 2018. A classifier ensemble for offensive text detection. Em *24th Brazilian Symposium on Multimedia and the Web*, 237–243. 10.1145/3243082.3243111

Pham, Chau, Alexander Hoyle, Simeng Sun, Philip Resnik & Mohit Iyyer. 2024. TopicGPT: A prompt-based topic modeling framework. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2956–2984. 10.18653/v1/2024.naacl-long.164

Pires, Ramon, Hugo Queiroz Abonizio, Thales Sales Almeida & Rodrigo Frassetto Nogueira. 2023. Sabiá: Portuguese large language models. Em *Intelligent Systems*, 226–240. 10.1007/978-3-031-45392-2_15

Plath, Hannah O., Maria Estela O. Paiva, Danielle L. Pinto & Paula D. P. Costa. 2022.

Detecção de discurso de Ódio contra mulheres em textos em português brasileiro: Construção da base MINA-BR e modelo de classificação. *Revista Eletrônica de Iniciação Científica em Computação* 20(3).

Rodrigues, João, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso & Tomás Osório. 2023. Advancing neural encoding of portuguese with transformer Albertina PT-*. arXiv [cs.CL]. 10.48550/arXiv.2305.06721

Santos, Raquel Bento, Bernardo Cunha Matos, Paula Carvalho, Fernando Batista & Ricardo Ribeiro. 2022. Semi-supervised annotation of Portuguese hate speech across social media domains. Em *11th Symposium on Languages, Applications and Technologies (SLATE)*, 11:1–11:14. 10.4230/OASIcs.SLATE.2022.11

Santos, Rodrigo, João Silva, Luís Gomes, João Rodrigues & António Branco. 2024. Advancing generative ai for Portuguese with open decoder Gervásio PT-*. arXiv [cs.CL]. 10.48550/arXiv.2402.18766

Saraiva, Ghivvago Damas, Rafael Anchiêta, Francisco Assis Ricarte Neto & Raimundo Moura. 2021. A semi-supervised approach to detect toxic comments. Em *International Conference on Recent Advances in Natural Language Processing (RANLP)*, 1261–1267.

Seno, Eloize Rossi Marques, Daniela Claro, Laila Mota & Jessica Rodrigues. 2024. Semântica distribucional. Em *Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português*, BPLN 2nd edn.

da Silva, Rodolfo Costa Cezar & Thierson Couto Rosa. 2023. Combining data transformation and classification approaches for hate speech detection: A comparative study. SSRN Preprint. 10.2139/ssrn.4477182

Silva, Rodolvo, Deborah Fernandes & Márcio Fernandes. 2018. Caracterização de mensagens em língua portuguesa com traços de racismo no Twitter. Em *6th Escola Regional de Informática de Goiás*, 205–214.

Souza, Andrey, Eduardo Nakamura & Fabíola Nakamura. 2022. Detecção de discurso de Ódio: Homofobia. Em *16th Brazilian e-Science Workshop*, 73–80. 10.5753/bresci.2022.223222

Souza, Fábio, Rodrigo Nogueira & Roberto Lotufo. 2020. BERTimbau: Pretrained BERT models for Brazilian Portuguese. Em *Intelligent Systems*, 403–417. 10.1007/978-3-030-61377-8_28

de Souza, Lucas, Franciele Beal, André Ortoncelli & Marlon Marcon. 2024. Detection and censorship of offensive language in extended texts in Portuguese. Em *15th XV Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, 202–211. 10.5753/stil.2024.245421

Thorndike, Robert L. 1953. Who belongs in the family? *Psychometrika* 18(4). 267–276. 10.1007/BF02289263

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave & Guillaume Lample. 2023a. LLaMA: Open and efficient foundation language models. arXiv [cs.CL]. 10.48550/arXiv.2302.13971

Touvron, Hugo, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov & Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. arXiv [cs.CL]. 10.48550/arXiv.2307.09288

Trajano, Douglas, Rafael H. Bordini & Renata Vieira. 2023. OLID-BR: offensive language identification dataset for Brazilian Portuguese. *Language Resources and Evaluation* 58. 1263–1289. 10.1007/s10579-023-09657-0

Vargas, Francielle, Isabelle Carvalho, Fabiana Rodrigues de Góes, Thiago Pardo & Fabrício Benevenuto. 2022. HateBR: A large expert

annotated corpus of Brazilian Instagram comments for offensive language and hate speech detection. Em *Language Resources and Evaluation Conference (LREC)*, 7174–7183.

Vargas, Francielle, Fabiana Rodrigues de Góes, Isabelle Carvalho, Fabrício Benevenuto & Thiago Pardo. 2021. Contextual-lexicon approach for abusive language detection. Em *Conference on Recent Advances in Natural Language Processing (RANLP)*, 1438–1447.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser & Illia Polosukhin. 2017. Attention is all you need. Em *31st Conference on Neural Information Processing Systems (NIPS)*,

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest & Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 38–45. 10.18653/v1/2020.emnlp-demos.6

Yin, Wenjie & Arkaitz Zubiaga. 2021. Towards generalisable hate speech detection: a review on obstacles and solutions. *PeerJ Computing Science* 7. e598. 10.7717/PEERJ-CS.598

Yosinski, Jason, Jeff Clune, Yoshua Bengio & Hod Lipson. 2014. How transferable are features in deep neural networks? Em *27th International Conference on Neural Information Processing Systems (NIPS)*, 3320–3328.

Apêndices

A. *Prompt one-shot*

Classifique o texto de rede social como "discurso de ódio" ou "ofensivo" ou "neutro".

Texto: Eu quero que ele vá pra puta que pariu , tô de saco cheio já .
Classe: ofensivo

Texto: {texto_alvo}
Classe:

B. *Prompt* com contexto adicional

Classifique o texto de rede social como "discurso de ódio", "ofensivo" ou "neutro".

Considerando que os assuntos da classe "discurso de ódio" estão associados com as palavras e emojis: mulher, bonito, bixa, viado, amo, ninguém, flor, koe, doido, boiola.
Da classe "ofensivo" estão associados com as palavras e emojis: pessoa, aborto, macho, presidente, mito, xis, podre, caneta, :rosto_sorridente_com_chifres:, pau.
Da classe "neutro" estão associados com as palavras e emojis: ano, kakakaakk, mesmo, alguém, garota, pro, conheço, óbvio, versão, kkkk.

Texto: @user eu tambem, olha a jules falando que ama a rue
Classe: neutro

Texto: {texto_alvo}
Classe:

Proposta recebida em Setembro 2024 e aceite para publicação em Novembro 2024.

Avaliação Automática do Nível de Complexidade de Textos em Português Europeu

Automatic Assessment of Text Complexity Levels in European Portuguese

Eugénio Ribeiro
INESC-ID Lisboa, Portugal
Instituto Universitário de Lisboa (ISCTE-IUL), Portugal

Nuno Mamede
INESC-ID Lisboa, Portugal
Instituto Superior Técnico, Universidade de Lisboa, Portugal

Jorge Baptista
INESC-ID Lisboa, Portugal
Faculdade de Ciências Humanas e Sociais, Universidade do Algarve, Portugal

Resumo

A avaliação da inteligibilidade de textos e a sua classificação por níveis de complexidade é essencial para o ensino de língua e para indústrias relacionadas com a linguagem que dependem de uma comunicação eficaz. O Quadro Europeu Comum de Referência para as Línguas (CEFR) é uma referência amplamente reconhecida para a classificação dos níveis de proficiência linguística. Este quadro pode ser utilizado não apenas para avaliar a proficiência de aprendentes de uma língua, mas também, de uma perspetiva de inteligibilidade, como um meio de identificar a proficiência necessária para compreender um texto. O objetivo deste estudo é desenvolver e avaliar modelos automáticos capazes de classificar textos em português europeu de acordo com os níveis de complexidade definidos pelo CEFR. Para tal, exploramos o ajuste de vários modelos de base pré-treinados em dados textuais utilizados para fins de avaliação de proficiência e exploramos abordagens que tiram partido da natureza ordinal dos níveis. Realizamos ainda uma análise preliminar da capacidade de base que modelos baseados em instruções têm para desempenhar esta tarefa. Nas experiências, os melhores modelos conseguem atingir mais de 80% de taxa de acerto e 75% de medida F_1 mas têm dificuldade em generalizar para diferentes tipos de texto, o que revela a necessidade de dados de treino adicionais e mais diversificados.

Palavras chave

inteligibilidade; complexidade textual; português europeu

Abstract

The assessment of text readability and the classification of texts by complexity levels is essential for language education and language-related industries that rely on effective communication. The Common European Framework of Reference for Languages (CEFR) provides a widely recognized framework for classifying language proficiency levels. This framework can be used not only to assess the proficiency of learners of a given language, but also from a readability perspective, as a means to identify the proficiency required to understand specific pieces of text. This study aims to develop and evaluate automatic models capable of classifying texts in European Portuguese according to the complexity levels defined by the CEFR. For that, we explore the fine-tuning of several foundation models on textual data used for proficiency evaluation purposes. Additionally, we explore approaches to leverage the information provided by the ordinal nature of the levels. Furthermore, we perform a preliminary analysis of the base capability of instruction-based models to perform this task. Our experiments show that the best models can achieve over 80% accuracy and 75% F_1 score. However, they have difficulty in generalizing to different types of text, which reveals the need for additional and more diverse training data.

Keywords

readability; text complexity; European Portuguese

DOI: 10.21814/lm.16.2.449
This work is Licensed under a Creative Commons Attribution 4.0 License

1. Introdução

Identificar o nível de inteligibilidade, complexidade, ou dificuldade de um texto é relevante em diversos domínios, abrangendo não só a aprendizagem de línguas, mas também várias indústrias relacionadas com a linguagem e muitas outras atividades humanas. Num contexto educacional, avaliar corretamente o nível de complexidade permite que educadores escolham textos adequados às capacidades dos alunos, promovendo um desenvolvimento linguístico eficaz e experiências de aprendizagem personalizadas. Além disso, fora do domínio da educação, a classificação do nível de inteligibilidade encontra aplicações em diferentes setores. Por exemplo, na indústria bancária, apresentar informações e políticas financeiras com um nível de complexidade apropriado garante que os clientes possam entender os termos e condições, levando à toma de decisões bem informadas. Da mesma forma, na área da saúde, é crucial para indivíduos com diferentes níveis de proficiência na língua que as instruções médicas, formulários de consentimento e materiais de informação sejam acessíveis e compreensíveis. Além disso, informações legais, comunicações governamentais e manuais de utilizador, entre muitos outros, podem beneficiar de uma avaliação adequada do nível de inteligibilidade, facilitando a comunicação, a transparência do conteúdo e a compreensão em geral.

No contexto da educação, o Quadro Europeu Comum de Referência para as Línguas (CEFR) (Council of Europe, 2001) define um enquadramento amplamente reconhecido para classificar níveis de proficiência linguística, variando entre A1 (iniciante) e C2 (proficiente). Além disso, tem duas adaptações específicas para o português europeu: o Quadro de Referência para o Ensino Português no Estrangeiro (QuaREPE) (Grosso et al., 2011) e o Referencial Camões PLE (Direção de Serviços de Língua e Cultura, 2017). Embora este quadro seja tipicamente usado para avaliar o nível de proficiência dos aprendentes de uma determinada língua, ele também pode ser visto de uma perspetiva de inteligibilidade, como um indicador do nível de proficiência necessário para entender um determinado texto (Hulstijn, 2007, 2011). Deste modo, explorando a perspetiva de inteligibilidade do CEFR, é possível analisar os fatores que influenciam a compreensão do texto e as suas implicações para a educação e para as indústrias relacionadas com a língua que procuram transmitir informações a aprendentes ou clientes de forma clara, concisa e de fácil compreensão.

Determinar o nível de complexidade ou inteligibilidade de textos apresenta seu próprio conjunto de desafios, especialmente ao trabalhar com línguas para as quais a quantidade de recursos anotados é limitada. Anotar grandes quantidades de dados de texto com os níveis definidos pelo CEFR é uma tarefa intensiva, demorada e que, na maioria das vezes, exige conhecimento especializado do domínio. Consequentemente, existe uma escassez de dados anotados, o que dificulta o desenvolvimento de modelos robustos e precisos para a classificação automática do nível de complexidade em múltiplos idiomas. Ainda assim, neste estudo, abordamos essa tarefa no contexto do português europeu.

Neste contexto, os dados usados na maioria dos estudos anteriores foram obtidos através da extração de textos usados nos exames para avaliação de proficiência pelo Camões, I.P.[1], o instituto oficial para a promoção da língua portuguesa. No entanto, esses dados não estão disponíveis publicamente, variam ao longo do tempo e, como demonstrado em estudos anteriores (Branco et al., 2014a; Curto, 2014), existem alguns problemas ao nível da classificação. Numa tentativa de mitigar alguns destes problemas, definimos um novo conjunto de teste com base nos exames modelo disponibilizados publicamente na página do instituto (Ribeiro et al., 2024a). Neste estudo analisamos este conjunto de teste em mais detalhe, inclusive através da reanotação do nível de complexidade dos textos por peritos.

Neste estudo, o nosso objetivo principal é investigar métodos para a classificação automática do nível de complexidade de textos em português europeu, fornecendo uma abordagem para a avaliação automática da inteligibilidade textual. Para alcançar este objetivo, neste artigo, agregamos e estendemos as análises dos nossos artigos publicados nas atas da conferência PROPOR 2024 (Ribeiro et al., 2024a,b), em que comparamos o desempenho de vários modelos de base existentes para o português quando ajustados para esta tarefa, assim como o desempenho de diferentes abordagens para capturar a relação ordinal entre os vários níveis. Além disso, tendo em conta a proliferação de modelos generativos baseados em instruções e o seu desempenho em várias tarefas de Processamento de Língua Natural (PLN) (Ouyang et al., 2022), fazemos também uma análise do desempenho destes modelos nesta tarefa e num contexto *zero-shot*.

Neste artigo, começamos por dar, na Secção 2, uma visão geral dos trabalhos relacionados sobre a avaliação automática do nível de comple-

[1] https://www.instituto-camoes.pt/

xidade de textos, com foco no português europeu. Em seguida, na Secção 3, analisamos o conjunto de dados extraído dos exames do Camões, I.P., com especial detalhe no novo conjunto de teste. Na Secção 4, descrevemos as abordagens que exploramos para prever automaticamente o nível de complexidade. A Secção 5 descreve a configuração experimental, incluindo as metodologias de avaliação. Em seguida, na Secção 6, apresentamos e discutimos os resultados das experiências, incluindo os erros e enviesamentos observados para os diferentes modelos. Por fim, na Secção 7, resumimos as contribuições deste estudo, discutimos as suas limitações e fornecemos diretrizes para investigação futura.

2. Trabalho Relacionado

A avaliação da inteligibilidade ou complexidade de textos é um problema que tem sido amplamente explorado ao longo dos anos (Leal & Aluísio, 2024). Tradicionalmente, o problema é abordado através da criação de fórmulas ou índices de inteligibilidade com base em informações estatísticas e/ou conhecimento de domínio (DuBay, 2004; Crossley et al., 2017). Entre estes, os mais usados são o Índice de Facilidade de Leitura de Flesch e o Índice de Legibilidade de Flesch-Kincaid (Kincaid et al., 1975).

No entanto, considerando os desenvolvimentos em aprendizagem automática, e especialmente em PLN, a investigação em avaliação automática da inteligibilidade e em tarefas relacionadas, como a avaliação da complexidade lexical (North et al., 2023), começou também a seguir as tendências na área de PLN. As primeiras abordagens (e muitas das iniciativas recentes para línguas com recursos limitados) basearam-se em características selecionadas manualmente, como a frequência de palavras, comprimento de frases e complexidade sintática, combinadas com algoritmos clássicos de aprendizagem automática, como árvores de decisão e Máquinas de Vetores de Suporte (SVM) (e.g., Aluisio et al., 2010; François & Fairon, 2012; Karpov et al., 2014; Curto et al., 2015; Pilán & Volodina, 2018; Forti et al., 2020; Leal et al., 2023). Posteriormente, surgiram abordagens de aprendizagem profunda baseadas em *embeddings* de palavras pré-treinados, como os gerados usando Word2Vec (Mikolov et al., 2013) (e.g., Cha et al., 2017; Nadeem & Ostendorf, 2018; Filighera et al., 2019). Por fim, mais recentemente, a investigação na área mudou-se para o ajuste de modelos pré-treinados baseados em transformadores, como BERT (Devlin et al., 2019), GPT (Radford et al., 2019) e RoBERTa (Liu et al., 2019); por exemplo, (Santos et al., 2021; Yancey et al., 2021; Martinc et al., 2021; Mohtaj et al., 2022).

Embora vários estudos tenham abordado a avaliação da inteligibilidade ou complexidade de textos como uma tarefa de regressão (e.g., Marujo et al., 2009; Cha et al., 2017; Nadeem & Ostendorf, 2018; Martinc et al., 2021; Wilkens et al., 2022; Mohtaj et al., 2022), apenas alguns exploraram as diferenças entre abordagens de regressão e classificação para a tarefa. Heilman et al. (2008) compararam regressão linear, regressão logística ordinal (McCullagh, 1980) e regressão logística multiclasse. A segunda abordagem obteve o melhor desempenho em termos de correlação, erro quadrático médio e taxa de acerto adjacente num cenário de validação cruzada. Contudo, o modelo de regressão linear mais simples revelou maior capacidade de generalização. Aluisio et al. (2010) compararam o desempenho de SVMs treinadas para classificação, regressão e classificação ordinal. Os modelos revelaram desempenho semelhante, mas cada um teve uma leve vantagem em termos de uma das métricas de avaliação, com a classificação a alcançar o maior valor de F_1, a regressão a maior correlação e a classificação ordinal o menor erro. Por fim, Xia et al. (2016) compararam uma abordagem de classificação usando SVMs com uma abordagem baseada em *ranking* de pares e alcançaram uma melhor correlação com a abordagem de classificação.

Tal como para a maioria das tarefas de PLN, uma parte significativa da investigação sobre avaliação automática do nível de inteligibilidade de texto foca-se na língua inglesa (e.g., Xia et al., 2016; Cha et al., 2017; Nadeem & Ostendorf, 2018; Filighera et al., 2019; Martinc et al., 2021). No entanto, neste caso, também existem vários estudos que abordam o problema em outras línguas, muitas das quais com recursos limitados. Por exemplo, existem estudos em francês (e.g., François & Fairon, 2012; François et al., 2020; Yancey et al., 2021; Wilkens et al., 2022; Hernandez et al., 2022), chinês (e.g., Sung et al., 2015), alemão (e.g., Mohtaj et al., 2022), italiano (e.g., Forti et al., 2020; Santucci et al., 2020), russo (e.g., Karpov et al., 2014; Reynolds, 2016), sueco (e.g., Jönsson et al., 2018; Pilán & Volodina, 2018) e esloveno (e.g., Martinc et al., 2021).

Focando no português, embora existam alguns estudos sobre a variedade brasileira da língua (e.g., Scarton & Aluísio, 2010; Aluisio et al., 2010; Leal et al., 2023), neste estudo estamos principalmente interessados na variedade europeia. Assim, apresentamos abaixo com mais detalhe estudos anteriores que cobrem esta variedade.

A versão portuguesa do sistema de tutoria REAP (Marujo et al., 2009) incluía um classificador de nível de inteligibilidade treinado em manuais escolares do 5º ao 12º ano. O modelo baseava-se em SVMs aplicados a características lexicais, como estatísticas de unigramas de palavras, e usava regressão logística ordinal para capturar a natureza ordinal dos níveis. Embora este modelo fosse preciso quando aplicado a manuais escolares, o seu desempenho diminuiu significativamente quando aplicado a exames do 6º, 9º e 12º ano de escolaridade.

A ferramenta LX-CEFR (Branco et al., 2014b) foi desenhada para ajudar aprendentes e professores de português a avaliar o nível de complexidade de um texto de acordo com o CEFR. A ferramenta foca-se em quatro características de forma independente: o Índice de Facilidade de Leitura de Flesch, a densidade lexical na forma da proporção de substantivos, o comprimento médio das palavras em número de sílabas e o comprimento médio das frases em número de palavras. Um corpus de 114 excertos extraídos dos exames de português como língua estrangeira realizados pelo Camões, I.P., foi usado para calcular a correlação entre estas características e o nível de inteligibilidade. Um estudo subsequente (Branco et al., 2014a) focou-se na reavaliação da ferramenta por especialistas humanos, assim como na reanotação dos textos por múltiplos instrutores de línguas. Neste caso, a concordância entre anotadores foi de apenas 0,17, o que revela a dificuldade e subjetividade da tarefa.

Curto et al. (2015) exploraram o uso de vários algoritmos clássicos de aprendizagem automática para a tarefa. Os algoritmos foram aplicados sobre 52 características divididas em 5 grupos diferentes: partes do discurso, blocos, frases e palavras, verbos, médias e frequências, e extras. As experiências foram realizadas numa versão estendida do conjunto de dados usado no contexto da ferramenta LX-CEFR, contendo 237 excertos. O melhor desempenho foi alcançado usando o algoritmo LogitBoost (Friedman et al., 2000). As características que se revelaram mais importantes foram as relacionadas com o tamanho dos textos em número de palavras e frases, o número de palavras diferentes, a quantidade de dependências sintáticas e o número de nós na árvore sintática. Além disso, à semelhança do que foi observado por Branco et al. (2014a), uma reanotação desta versão estendida do conjunto de dados por dois grupos de múltiplos especialistas revelou valores baixos para a concordância entre anotadores: 0,19 e 0,16 (Curto, 2014).

Akef et al. (2024) também exploraram o uso de algoritmos clássicos de aprendizagem, desta vez aplicados sobre um conjunto de 489 características separadas em 6 grupos: contagens, características lexicais, sintáticas, discursivas, morfológicas e psicolinguísticas. Além disso, os modelos foram treinados em múltiplas variantes do conjunto de dados extraído dos exames do Camões, I.P., incluindo uma com 500 excertos. O desempenho foi semelhante ao atingido por Curto et al. (2015). Contudo, experiências adicionais mostraram que os modelos têm pouca capacidade de generalização para um conjunto de dados composto por textos extraídos de livros para aprendizagem de português como segunda língua.

Correia & Mendes (2021) exploraram o uso de uma rede neuronal híbrida com dois ramos, um deles focado no processamento bidirecional de *embeddings* de palavras pré-treinados (NILC-Embeddings (Hartmann et al., 2017)) e o outro num conjunto de 14 características baseadas em comprimento e índices de inteligibilidade. A aplicação desta rede neuronal sobre o conjunto de dados do Camões, I.P., com 500 excertos mostrou que a combinação dos dois ramos leva a um desempenho superior àquele atingido por cada ramo individualmente.

Por fim, passando para o uso de abordagens baseadas no ajuste de modelos pré-treinados, Santos et al. (2021) exploraram o ajuste de versões em português dos modelos GPT-2 (Radford et al., 2019) e RoBERTa (Liu et al., 2019) em múltiplas variantes do conjunto de dados extraído dos exames do Camões, I.P.. Em geral, nas versões maiores do conjunto de dados, estas abordagens atingiram melhor desempenho que as abordagens clássicas, sendo os melhores resultados obtidos usando o modelo GPT-2. Akef et al. (2024) também exploraram o uso de uma versão ajustada do modelo GPT-3.5-Turbo, reportando um desempenho superior ao do modelo GPT-2, embora com uma metodologia de avaliação diferente. Nos nossos estudos (Ribeiro et al., 2024a,b), explorámos o uso de vários modelos de base adicionais, atingindo os melhores resultados usando um modelo da família Albertina PT-* (Rodrigues et al., 2023) e uma abordagem que tem em conta a relação ordinal entre os vários níveis. Como este artigo descreve uma extensão desses estudos, os resultados concretos serão apresentados e discutidos mais à frente. Relativamente aos resultados dos restantes estudos, tendo em conta as variações no conjunto de dados e na metodologia de avaliação, é difícil fazer uma comparação que cubra todas as abordagens. Ainda assim, a Tabela 1 sumariza os resultados

Abordagem	**Taxa de Acerto**	**Macro F_1**
LogitBoost (Curto et al., 2015)	68,60	64,30
LogitBoost (Akef et al., 2024)	68,00	58,66
Floresta Aleatória (Akef et al., 2024)	72,00	61,17
Rede Neuronal Híbrida (Correia & Mendes, 2021)	73,00	-
GPT-2 (Santos et al., 2021)	75,62	68,90
RoBERTa (Santos et al., 2021)	72,50	58,90
GPT-3.5-Turbo (Akef et al., 2024)	79,00	70,11

Tabela 1: Resultados obtidos em estudos anteriores sobre o conjunto de dados com 500 excertos extraídos dos exames do Camões, I.P.. Os resultados estão na forma de percentagem. Os resultados da abordagem de Curto et al. (2015) sobre este conjunto de dados foram obtidos por Santos et al. (2021). Os resultados do modelo GPT-3.5-Turbo foram obtidos sobre uma partição de teste enquanto os das restantes abordagens foram obtidos usando validação cruzada.

obtidos pelas abordagens aplicadas sobre o conjunto de dados com 500 excertos extraídos dos exames do Camões, I.P., em termos de taxa de acerto e medida macro F_1.

3. Conjunto de Dados

De modo semelhante aos estudos anteriores sobre a análise automática da complexidade de textos em português europeu discutidos na Secção 2, o nosso conjunto de dados é composto por textos extraídos dos exames para avaliação de proficiência realizados pelo Camões, I.P.[2], o instituto oficial de promoção da língua portuguesa. Os textos abrangem os níveis A1 a C1 do CEFR, conforme definidos na versão portuguesa do quadro (Grosso et al., 2011; Direção de Serviços de Língua e Cultura, 2017). Tendo em conta que estes textos são utilizados para fins de avaliação e podem ser reutilizados ao longo do tempo, eles não estão disponíveis publicamente, o que dificulta a investigação sobre esta tarefa. Por uma questão de simplicidade, doravante chamaremos estes exames de exames privados. Além disso, o número de textos anotados aumenta ao longo do tempo e não existe uma partição padrão dos dados, o que levou à utilização de múltiplas versões diferentes do conjunto de dados nos estudos anteriores. Como tal, é difícil comparar as abordagens existentes sem repetir as experiências. Por fim, alguns estudos (Branco et al., 2014a; Curto, 2014) demonstraram que existe discrepância entre os níveis dos textos e os que lhe são atribuídos por anotadores humanos. Em parte, esta discrepância pode ser explicada pela inferência do nível dos textos a partir do nível do exame do qual são extraídos. Uma vez que, no contexto dos exames, cada texto está associado a uma determinada tarefa a realizar, este nível é apenas uma aproximação. Ainda assim, a baixa concordância entre anotadores observada em estudos anteriores revela que parte da discrepância também se deve à subjetividade inerente à definição do nível de complexidade dos textos.

Numa tentativa de mitigar alguns destes problemas, definimos um novo conjunto de teste[3] com base nos textos incluídos nos exames modelo (um para cada nível) disponibilizados na página do instituto. Uma vez que estes exames estão disponíveis publicamente, este conjunto de teste pode ser usado para padronizar a avaliação, permitindo que investigadores sem acesso aos exames privados possam pelo menos avaliar as suas abordagens. Inicialmente, tal como para os textos extraídos dos exames privados, usamos o nível do exame como uma aproximação do nível dos textos. No entanto, tendo em conta a discrepância entre esta aproximação e o nível atribuído por anotadores humanos observada em estudos anteriores, optámos por também recorrer a peritos para fazer a anotação dos textos. A concordância (κ de Fleiss) entre os três anotadores foi de apenas 0,21, o que confirma a subjetividade e a dificuldade da tarefa. A concordância (κ de Cohen) entre o nível do exame e o nível médio atribuído pelos anotadores é de 0,29.

A Tabela 2 mostra a distribuição dos textos pelos níveis do CEFR. Aquando da realização deste estudo, havia 598 textos extraídos dos exames que não estão disponíveis publicamente. Podemos ver que há um viés na direção do nível central (B1) e um número menor de exemplos dos níveis mais avançados. Além disso, considerando que alguns dos textos são reutilizados em

[2] https://www.instituto-camoes.pt/

[3] https://gitlab.hlt.inesc-id.pt/iread4skills/cmodel

	A1	A2	B1	B2	C1	Total
Treino	92	157	240	49	60	598
Teste (E)	8	12	5	3	4	32
Teste (A)	6	14	4	7	1	32

Tabela 2: Distribuição dos textos extraídos dos exames do Camões, I.P., pelos níveis do CEFR. As anotações (E) e (A) associadas ao conjunto de teste correspondem ao nível do exame e ao nível médio atribuído pelos anotadores, respetivamente.

vários exames, alguns dos exemplos consistem em pequenas variações do mesmo texto.

Relativamente ao conjunto de teste extraído dos exames modelo, podemos ver que este consiste em apenas 32 textos, o que introduz algumas limitações. Os textos cobrem vários tipos, como excertos de narrativas, diálogos, notícias e cartas. O Apêndice A inclui um exemplo de cada nível, selecionado de entre os casos em que existe concordância entre o nível do exame e o atribuído pelos anotadores. Neste contexto, analisando as distribuições na Tabela 2, podemos ver que existe uma tendência dos anotadores para evitar o nível C1. No entanto, em geral, a variação entre as duas classificações foi equilibrada (7 textos considerados de um nível mais fácil pelos anotadores e 8 de um nível mais avançado) e em apenas um caso foi superior a um nível. Em comparação com a distribuição dos textos dos exames privados, podemos ver que no conjunto de teste existe uma prevalência maior dos níveis A. Isto deve-se a um tipo de exercício de compreensão escrita que envolve múltiplos textos curtos e que só ocorre nos exames desses níveis. A Tabela 3 mostra alguns exemplos deste tipo de textos. A diferença em relação à distribuição dos textos extraídos dos exames privados deve-se à não inclusão deste tipo de textos no conjunto disponibilizado, que consiste apenas em textos mais longos. Como tal, o conjunto de teste extraído dos exames modelo serve também para avaliar a capacidade de generalização dos modelos para tipos de texto não observados durante o treino.

4. Abordagens

Analisando os resultados dos estudos anteriores sobre a avaliação automática do nível de complexidade de textos referidos na Secção 2, chegamos à conclusão de que, atualmente, o melhor desempenho na tarefa é atingido através do ajuste de modelos de base pré-treinados (Bommasani et al., 2021). Como tal, o principal objetivo deste estudo é comparar o desempenho dos diversos modelos de base existentes para o português quando ajustados para esta tarefa. No entanto, tendo em conta a proliferação de modelos generativos baseados em instruções e o seu desempenho em várias tarefas de PLN (Ouyang et al., 2022), fazemos também uma análise preliminar do desempenho destes modelos para esta tarefa.

4.1. Ajuste de Modelos Pré-Treinados

O ajuste de modelos pré-treinados, conhecido como *fine-tuning*, é uma técnica na qual um modelo previamente treinado numa grande quantidade de dados é adaptado para uma tarefa específica. Isso é feito continuando o treino do modelo, mas com um conjunto de dados mais especializado e um objetivo de tarefa particular, como a classificação de textos por nível de complexidade.

Apesar de os níveis do CEFR terem uma natureza ordinal, os estudos mais recentes sobre a avaliação automática do nível de complexidade de textos em português (Curto et al., 2015; Santos et al., 2021; Correia & Mendes, 2021; Akef et al., 2024) abordaram o problema como uma tarefa de classificação sem considerar as relações entre os níveis. Posto isto, neste estudo, além de analisar o desempenho dos vários modelos de base quando ajustados para a tarefa de classificação, exploramos também abordagens que tentam tirar partido da natureza ordinal dos níveis.

Em primeiro lugar, dado o mesmo modelo de base, comparamos o seu desempenho quando ajustado para a tarefa de classificação com o seu desempenho quando ajustado para uma tarefa de regressão. No primeiro caso, cada nível é considerado uma classe independente enquanto, no segundo, os níveis são convertidos em valores numéricos ordenados. No caso da classificação, o modelo produz uma aproximação da distribuição de probabilidade das classes usando a função *softmax*, sendo escolhida aquela com maior probabilidade. No caso da regressão, o modelo produz um valor numérico contínuo. Neste caso, a previsão do nível é obtida por arredondamento ao nível mais próximo.

Além disso, exploramos a adaptação *a posteriori* do modelo de classificação para ter em conta a ordem dos níveis usando duas abordagens distintas. A primeira consiste em calcular a média ponderada da aproximação da distribuição de probabilidades dada pela função *softmax*. A segunda consiste em usar essa distribuição de probabilidades para calcular a função de distribuição acumulada que, neste caso, representa a probabilidade de a complexidade de um determinado texto ser menor ou igual a um determinado nível. Em seguida, usamos essa função para treinar um modelo de regressão logística ordinal.

Exame	Anotadores	Exemplo
A1	A1	*É proibido sair da sala sem autorização do professor.*
A1	A2	*É favor não jogar à bola no interior da escola.*
A2	A1	*Avariado. Pedimos desculpa pelo incómodo.*
A2	A2	*Lamentamos mas não é possível atendê-lo agora. Tente mais tarde.*

Tabela 3: Exemplos de textos curtos que ocorrem apenas nos exames dos níveis A.

4.2. Modelos Baseados em Instruções

Recentemente, temos observado a proliferação de modelos generativos baseados em instruções (Ouyang et al., 2022), tendo o ChatGPT (OpenAI, 2023) como principal impulsionador. Estes modelos têm revelado um desempenho satisfatório em várias tarefas de PLN, especialmente em contextos com poucos ou mesmo nenhuns exemplos. Logo, é importante avaliar o desempenho destes modelos na análise do nível de complexidade de textos.

Neste estudo, focamo-nos apenas num cenário *zero-shot*, ou seja, não são fornecidos quaisquer exemplos ao sistema. Deste modo, analisamos a capacidade de base destes modelos e o seu conhecimento intrínseco para desempenhar esta tarefa. Os cenários que envolvem exemplos introduzem níveis de complexidade adicionais, relacionados com a escolha dos exemplos a fornecer e com as limitações ao nível do tamanho da janela de contexto que os modelos são capazes de processar. Como tal, deixamos a exploração desses cenários para trabalho futuro.

O desempenho dos modelos baseados em instruções é significativamente influenciado pela qualidade das instruções (Mishra et al., 2022; Giray, 2023; Chen et al., 2023). Neste contexto, exploramos o uso de duas instruções diferentes:

Instrução A Classifica o nível de complexidade do seguinte texto de acordo com o Quadro Europeu Comum de Referência para as Línguas (CEFR). Responde apenas com o nível: A1, A2, B1, B2, C1.

Instrução B De acordo com o Quadro Europeu Comum de Referência para as Línguas (CEFR), qual é o nível de proficiência necessário para compreender o seguinte texto? Responde apenas com o nível: A1, A2, B1, B2, C1.

Enquanto a primeira instrução aponta diretamente ao nível de complexidade do texto, a segunda aborda o problema do ponto de vista do nível de proficiência necessário para compreender o texto.

5. Configuração Experimental

Esta secção descreve a nossa configuração experimental, começando com a abordagem usada como linha de base para as nossas experiências na Secção 5.1. Em seguida, descrevemos os modelos usados nas nossas abordagens na Secção 5.2. A Secção 5.3 descreve a metodologia de avaliação. Por fim, a Secção 5.4 fornece mais alguns detalhes de implementação para aumentar a reprodutibilidade das nossas experiências.

5.1. Linha de Base

Como linha de base usamos um classificador baseado em *gradient boosting* (Friedman, 2001) aplicado a um conjunto de 631 características descritivas, lexicais, sintáticas e discursivas, que foram identificadas no contexto do projeto iRead4Skills[4] como potenciais indicadores da complexidade de um texto[5].

5.2. Modelos

As abordagens usadas no nosso estudo tiram partido de modelos existentes, seja para os ajustar à tarefa ou para os usar diretamente no caso da abordagem baseada em instruções. Nesta secção, descrevemos os vários modelos usados neste estudo, começando pelos modelos de base pré-treinados, na Secção 5.2.1, seguidos dos modelos baseados em instruções, na Secção 5.2.2.

5.2.1. Modelos de Base

Em termos de modelos de base pré-treinados (Bommasani et al., 2021), o nosso objetivo é cobrir extensivamente os modelos que estão atualmente disponíveis publicamente para o português, independentemente da variedade linguística (brasileira ou europeia). Estes modelos são descritos em seguida:

BERTimbau (Souza et al., 2020) é o modelo de base em português mais utilizado. Este segue

[4] https://iread4skills.com/

[5] https://gitlab.hlt.inesc-id.pt/iread4skills/docs

a arquitetura original do BERT (Devlin et al., 2019), mas foi treinado no Brazilian Web as a Corpus (brWaC) (Wagner Filho et al., 2018) exclusivamente para modelação de língua com *tokens* mascarados. Inicialmente foram treinadas duas variantes do modelo: *base* e *large* com 110M e 335M de parâmetros, respetivamente. Existe também uma versão destilada, obtida através da aplicação da abordagem DistilBERT (Sanh et al., 2019) sobre a variante *base*.

BERTugues (Zago, 2023) é um modelo que foi treinado com o objetivo de superar o BERTimbau. Este foi treinado numa versão filtrada do brWaC em que textos com pouca qualidade foram descartados. Além disso, também foi treinado para previsão da próxima frase. O seu tokenizador inclui *emojis* e descarta caracteres que ocorrem apenas muito raramente em português. Contudo, ao contrário do BERTimbau, o BERTugues só possui uma variante *base*, com 110M de parâmetros.

RoBERTa PT (Santos et al., 2021) é uma versão pequena do RoBERTa (Liu et al., 2019) com 68M de parâmetros, treinada em 10 milhões de frases em português e 10 milhões de frases em inglês do corpus OSCAR (Suárez et al., 2019). Foi treinado por Santos et al. (2021) para ser usado num estudo sobre avaliação automática do nível de inteligibilidade.

GPorTuguese-2 (Guillou, 2020) é uma versão ajustada do modelo GPT-2 *small* em inglês (Radford et al., 2019) na Wikipedia em português. Possui 124M de parâmetros. Este foi o modelo usado como base para alcançar o melhor desempenho no estudo sobre avaliação automática do nível de inteligibilidade de Santos et al. (2021).

Albertina PT-* (Rodrigues et al., 2023; Santos et al., 2024) é uma família de modelos baseados no DeBERTa (He et al., 2021). Existem modelos para o português europeu e para o português do Brasil. Para cada variedade linguística, existem três variantes, com 100M, 900M e 1.5B de parâmetros, respetivamente. Os modelos para português do Brasil foram treinados no brWaC, enquanto os modelos para português europeu foram treinados numa combinação de transcrições de debates no Parlamento Português, as partes em português dos corpora do Parlamento Europeu e a parte em português europeu do corpus OSCAR. Atualmente, estes modelos pré-treinados são os que atingem o melhor desempenho quando ajustados para múltiplas tarefas de PLN em português.

5.2.2. Modelos Baseados em Instruções

Relativamente aos modelos baseados em instruções, tal como descrito na Secção 4.2, o nosso objetivo é fazer um estudo preliminar da capacidade destes modelos para desempenhar a tarefa de classificação da complexidade de textos num cenário *zero-shot*. Tendo em conta que o ChatGPT (OpenAI, 2023) é o principal impulsionador do uso de modelos baseados em instruções, exploramos o uso de vários modelos da família GPT (Radford et al., 2018). No entanto, tendo em conta que esses modelos não são abertos, exploramos também o uso de modelos das famílias Llama (Touvron et al., 2023) e Mistral (Jiang et al., 2023).

GPT (Radford et al., 2018) é uma família de modelos que conta já com múltiplas gerações, todas elas com impacto significativo na área de PLN. A segunda geração desta família (Radford et al., 2019) já foi referida anteriormente como a base do modelo GPorTuguese-2 (Guillou, 2020). Porém, neste contexto, estamos interessados nos modelos baseados em instruções que surgiram após o desenvolvimento do InstructGPT (Ouyang et al., 2022) e que estão na base do ChatGPT (OpenAI, 2023). Mais especificamente, nas nossas experiências exploramos o uso dos seguintes modelos: GPT 3.5 Turbo, GPT 4 Turbo (OpenAI, 2024), GPT 4o e GPT 4o Mini. Embora se saiba que a capacidade geral destes modelos evoluiu entre gerações e que alguns dos modelos têm um tamanho mais reduzido (GPT 3.5 Turbo e GPT 4o Mini), os modelos são privados e, como tal, os detalhes da sua implementação não são conhecidos.

Llama (Touvron et al., 2023) é uma família de modelos disponíveis publicamente que foi desenvolvida para competir com os modelos da família GPT. A geração atual, Llama 3 (Llama Team, 2024), inclui modelos com 8B, 70B e 405B de parâmetros. Por uma questão de recursos, nas nossas experiências exploramos apenas o uso do modelo com 8B de parâmetros.

Mistral (Jiang et al., 2023) é uma família de modelos originalmente desenvolvidos para serem mais eficientes do que os da família Llama, atingindo um desempenho semelhante com um número menor de parâmetros. No entanto, a razão para explorarmos modelos desta família é o subconjunto de modelos Mixtral (Jiang et al., 2024), baseados em Mistura Esparsa de Especialistas (SMoE). Esta abordagem combina vários conjuntos de parâmetros (especialistas), sendo apenas um subconjunto deles selecionado em cada passo de geração. Existem dois modelos

deste tipo: Mixtral 8x7B e Mixtral 8x22B. Ambos são compostos por oito especialistas, sendo a diferença o número de parâmetros de cada especialista. Tal como para a família Llama, por uma questão de recursos, exploramos apenas o uso do modelo mais pequeno, com 7B de parâmetros por especialista.

5.3. Metodologia de Avaliação

Começando pelas métricas de avaliação, damos principal relevância à taxa de acerto (TA), à taxa de acerto adjacente (TAA) e à medida macro F_1, que são algumas das métricas mais comuns em estudos anteriores sobre classificação automática do nível de complexidade. A taxa de acerto avalia a identificação exata do nível de complexidade de um texto, enquanto a taxa de acerto adjacente também considera níveis vizinhos, fornecendo informação sobre a dispersão das classificações pelos níveis adjacentes. Considerando que a distribuição dos textos pelos níveis não é equilibrada, a medida macro F_1 também é uma métrica relevante para perceber se os classificadores tendem a favorecer as classes maioritárias. Adicionalmente, na parte do nosso estudo em que a tarefa é abordada como um problema de regressão, também analisamos a Raíz do Erro Quadrático Médio (REQM). Além disso, pontualmente, utilizamos valores de precisão e sensibilidade para analisar o desempenho para níveis específicos.

Os estudos sobre avaliação automática do nível de complexidade em Português Europeu descritos na Secção 2 utilizaram abordagens de validação cruzada para a avaliação. Como mencionado por Santos et al. (2021), a validação cruzada não é uma prática comum ao treinar modelos neuronais de grandes dimensões, pois é um processo demorado. Contudo, embora tenhamos definido um novo conjunto de teste para avaliação, também recorremos a uma abordagem de validação cruzada em 10 partes para ajustar os hiperparâmetros e identificar os modelos de base pré-treinados que levam ao melhor desempenho quando ajustados para a tarefa. Isto permite-nos avaliar o desempenho dos nossos modelos num cenário de avaliação semelhante ao dos estudos anteriores e, em seguida, usar o conjunto de teste para avaliar a capacidade de generalização dos modelos com melhor desempenho.

Em cada parte do processo de validação cruzada, os modelos de base são ajustados durante 20 épocas. Os pesos da melhor época são então selecionados de acordo com a taxa de acerto do modelo. Considerando que o processo de validação cruzada gera 10 modelos ajustados diferentes para cada modelo de base, usamos uma combinação destes modelos para gerar as previsões para o conjunto de teste. Para agregar as previsões dos vários modelos, experimentámos abordagens baseadas em probabilidade, ranking e votação por maioria. Não conseguimos identificar uma abordagem que fosse claramente melhor do que as outras. Por isso, optámos por fazer a média das distribuições de probabilidade das classes previstas pelos vários modelos.

Para aumentar a robustez e mitigar o impacto da aleatoriedade, realizámos múltiplas execuções experimentais independentes para todos os modelos. Algumas das experiências realizadas nos nossos estudos anteriores (Ribeiro et al., 2024a,b) consideraram 10 execuções. No entanto, por uma questão de consistência e devido à quantidade de recursos necessária para realizar 10 execuções de todas as experiências, optámos por considerar apenas 3 execuções neste estudo. Como tal, alguns dos resultados poderão ser ligeiramente diferentes. No caso do ajuste de modelos pré-treinados, cada execução usa uma semente aleatória diferente para o processo de partição da validação cruzada. No caso dos modelos baseados em instruções, uma vez que não é feito nenhum processo de treino adicional, estes são aplicados apenas ao conjunto de teste. Além disso, ajustamos os parâmetros disponíveis de forma a minimizar a variabilidade das respostas dos modelos entre diferentes execuções.

Salvo se indicado explicitamente, as métricas de avaliação são reportadas como a média e o desvio padrão das várias execuções. Todas as métricas exceto a REQM são reportadas em forma de percentagem.

5.4. Detalhes de Implementação

Para treinar o classificador baseado em características usado como linha de base, utilizamos a implementação do algoritmo *gradient boosting* fornecida pela biblioteca scikit-learn (Pedregosa et al., 2011).

Para ajustar os modelos pré-treinados, utilizamos a funcionalidade oferecida pela biblioteca Transformers da HuggingFace (Wolf et al., 2020). Usamos os valores padrão para a maioria dos hiperparâmetros, com a exceção do tamanho do lote (*batch size*) e a taxa de aprendizagem, para os quais experimentamos vários valores. Para a maioria dos modelos de base, os melhores resultados foram obtidos usando lotes de tamanho 32 e uma taxa de aprendizagem de 5×10^{-5}. Uma das exceções é o GPorTuguese-2, cujo desempenho é altamente influenciado pelo preenchimento (*padding*). Portanto, para maximizar esse desempenho, utilizamos lotes com apenas

um exemplo. Para além disso, os melhores resultados foram obtidos com uma taxa de aprendizagem de 1×10^{-5}. A outra exceção refere-se às versões maiores dos modelos Albertina PT-*, com 900M e 1.5B de parâmetros, que exibiram comportamento errático para valores maiores de ambos os hiperparâmetros. Por isso, utilizamos lotes de tamanho 16 e uma taxa de aprendizagem de 1×10^{-5}.

Para aplicar a regressão logística ordinal, usamos a implementação do fornecida pela biblioteca statsmodels (Seabold & Perktold, 2010).

Por fim, relativamente aos modelos baseados em instruções, usamos a API da OpenAI[6] para aceder aos modelos da família GPT. Para os modelos disponíveis publicamente, utilizamos a funcionalidade oferecida pela biblioteca Transformers da HuggingFace (Wolf et al., 2020) e as versões dos modelos treinadas especificamente para seguir instruções. Caso a resposta dos modelos inclua várias classificações, consideramos apenas a primeira.

6. Resultados

Na Secção 6.1 começamos por apresentar e analisar o desempenho dos vários modelos ajustados no cenário de validação cruzada. Em seguida, na Secção 6.2, avaliamos a capacidade de generalização dos melhores modelos, analisando o seu desempenho no conjunto de teste. Por fim, na Secção 6.3, usamos esse mesmo conjunto de teste para analisar o desempenho dos modelos baseados em instruções.

6.1. Validação Cruzada

A primeira linha da Tabela 4 mostra os resultados que usamos como linha de base no nosso estudo, obtidos através da aplicação do algoritmo *gradient boosting* a um conjunto de 631 características descritivas, lexicais, sintáticas e de discurso. Neste caso, atingimos uma taxa de acerto de 75,31%, superior à atingida por Curto et al. (2015) e Akef et al. (2024) num conjunto de dados mais reduzido. Isto revela a importância de ter uma maior quantidade de dados anotados para treinar os modelos.

As restantes linhas da Tabela 4 mostram os resultados obtidos ao ajustar os múltiplos modelos de base para esta tarefa. Primeiro, podemos ver que todos os modelos atingiram uma taxa de acerto acima de 75%, superando a linha de base. Tal como observado no estudo de Santos et al. (2021), isto mostra que, quando existe um número suficiente de exemplos de treino, o ajuste de modelos pré-treinados leva a um melhor desempenho que os modelos obtidos usando técnicas clássicas de aprendizagem automática. Além disso, o melhor modelo no estudo de Santos et al. (2021) obteve uma taxa de acerto de 75,62% na versão do conjunto de dados com 500 excertos. Mais uma vez, isto mostra que os dados de treino adicionais disponíveis para o nosso estudo têm um impacto significativo no desempenho dos modelos.

Analisando modelos específicos, começando pelo BERTimbau, o modelo pré-treinado mais utilizado para o português, podemos ver que o desempenho das suas três variantes é o esperado, com o modelo *large* superando o modelo *base*, e a versão destilada trocando menos de 1% de desempenho por um tamanho reduzido e tempos de treino e inferência mais rápidos.

O BERTugues conseguiu superar a versão *large* do BERTimbau, apesar de ter o mesmo número de parâmetros da versão *base*. Isto também foi observado por seu autor em outras tarefas de PLN em português (Zago, 2023) e revela as vantagens de treinar modelos de língua em dados com qualidade superior e de ter um tokenizador mais adequado à língua focada.

O RoBERTa PT, o modelo mais pequeno usado no nosso estudo, atingiu um desempenho semelhante ao da versão *large* do BERTimbau em termos de taxa de acerto e medida F_1 e o melhor desempenho global em termos de taxa de acerto adjacente. Isto pode ser justificado pelas melhorias no processo de treino utilizado pelo RoBERTa, como, por exemplo, o uso de máscaras dinâmicas (Liu et al., 2019). Além disso, é expectável que o pré-treino em frases em português do corpus OSCAR também contribua para o desempenho, pois faz com que a variedade europeia da língua seja considerada.

O GPorTuguese-2, o único modelo da família GPT (Radford et al., 2019) utilizado nesta parte do nosso estudo, é um dos que apresenta melhor desempenho, ocupando o segundo lugar em ¡todas as métricas. Tal como observado por Santos et al. (2021), este modelo superou o RoBERTa PT em termos de taxa de acerto (por dois pontos percentuais em comparação com três no estudo de Santos et al. (2021)). O desempenho obtido com este modelo sugere que ele ainda é uma escolha segura, apesar da existência de modelos de base pré-treinados mais recentes. No entanto, como o uso de preenchimento (*padding*) tem um grande impacto no seu desempenho, não é possível aproveitar ao máximo o hardware mo-

[6] https://openai.com/api/

Modelo	**TA**	**TAA**	**F_1**
Linha de Base	75,31±0,57	94,42±0,52	66,74±0,90
BERTimbau Large	79,26±2,09	95,99±0,61	71,68±2,61
BERTimbau Base	78,26±1,67	95,71±0,59	71,30±2,60
BERTimbau Distilled	77,65±0,68	95,71±0,51	70,98±0,60
BERTugues	79,43±0,29	95,54±0,51	72,76±0,77
RoBERTa PT	79,15±0,75	**97,05±0,25**	71,49±1,15
GPorTuguese-2	81,16±0,63	96,71±0,92	74,81±1,60
Albertina PT-PT 1.5B	75,69±0,86	93,26±0,10	68,51±1,16
Albertina PT-BR 1.5B	75,64±1,06	93,48±0,73	68,85±2,84
Albertina PT-PT 900M	77,42±0,34	94,48±0,67	70,92±0,65
Albertina PT-BR 900M	76,15±0,59	93,42±0,82	69,07±0,70
Albertina PT-PT 100M	**81,77±0,44**	96,27±0,54	**76,17±1,01**
Albertina PT-BR 100M	80,43±1,60	95,99±0,61	73,88±1,67

Tabela 4: Resultados obtidos pelos modelos ajustados no cenário de validação cruzada.

derno durante o ajuste deste modelo, tornando-o mais lento do que o ajuste da variante *large* do BERTimbau e quase tão lento quanto o ajuste dos modelos da família Albertina PT-* com um número de parâmetros nove vezes superior.

Analisando os resultados dos modelos da família Albertina PT-*, podemos ver que os modelos pré-treinados em dados em português europeu superam seus equivalentes em português do Brasil, embora as diferenças sejam menos significativas no caso dos modelos maiores. Isto confirma que as diferenças entre as duas variedades são relevantes e têm impacto na forma como o nível de dificuldade de um texto é percebido.

Além disso, entre os modelos dessa família, podemos encontrar os modelos quer com melhor quer com pior desempenho nesta tarefa. Os modelos maiores, com 900M e 1.5B de parâmetros, que alcançam desempenho de ponta em várias tarefas de PLN em português, foram os que obtiveram os piores resultados no nosso estudo em todas as métricas. No entanto, podemos argumentar que se trata de um caso de sobreajuste, pois esses modelos são demasiado grandes para o número de exemplos de treino disponíveis. Assim, é expectável que eles apresentem melhor desempenho com uma quantidade suficientemente grande e representativa de dados de treino. Por outro lado, os modelos com 100M de parâmetros estão entre os que apresentam melhor desempenho na tarefa, alcançando uma taxa de acerto acima de 80%.

Globalmente, o melhor desempenho no cenário de validação cruzada foi obtido ao ajustar a versão do modelo Albertina PT-PT com 100M de parâmetros. A taxa de acerto foi de 81,77% e a medida F_1 foi de 76,17%. Isto também representa a menor diferença entre as duas métricas entre todos os modelos. A este respeito, Santos et al. (2021) observaram uma diferença de 13,60 pontos percentuais ao usar o RoBERTa PT e de 6,72 pontos percentuais ao usar o GPorTuguese-2. Esses valores foram reduzidos para 7,66 e 6,35 no nosso estudo, o que sugere que os dados de treino adicionais levam a modelos menos tendenciosos. Ainda assim, a diferença entre as métricas sugere que os modelos ainda são um pouco tendenciosos ou que, pelo menos, têm mais dificuldade em identificar exemplos de certos níveis.

A Tabela 5 mostra as matrizes de confusão das melhores execuções dos dois modelos com melhor desempenho em comparação com linha de base. Podemos ver que em todos os casos, a sensibilidade é de aproximadamente 90% para o nível B1, que é simultaneamente o nível intermédio e o mais proeminente no conjunto de dados de treino. Por outro lado, os modelos parecem ter algumas dificuldades em distinguir entre os níveis A. A principal diferença entre os dois melhores modelos parece ser a forma como eles lidam com os níveis avançados. Enquanto o GPorTuguese-2 parece ter algumas dificuldades em distinguir entre os níveis B2 e C1, o Albertina PT-PT, com 100M de parâmetros, parece ser mais tendencioso em direção à previsão do nível B1.

Passando para a comparação entre as abordagens de classificação e de regressão, as duas primeiras linhas da Tabela 6 mostram os resultados obtidos por versões do modelo Albertina PT-PT 100M ajustadas para cada uma das visões sobre o problema. As duas linhas restantes mostram os resultados das adaptações do modelo de classificação para tentar capturar a relação ordinal entre os níveis: a média ponderada da aproximação da distribuição de probabilidade dada

		Linha de Base					Albertina PT-PT 100M						GPorTuguese-2				
		Previsto						Previsto						Previsto			
		A1	**A2**	**B1**	**B2**	**C1**		**A1**	**A2**	**B1**	**B2**	**C1**		**A1**	**A2**	**B1**	**B2**	**C1**
Real	**A1**	62	26	4	0	0	**Real** **A1**	73	16	3	0	0	**Real** **A1**	73	17	2	0	0
	A2	18	125	14	0	0	**A2**	22	133	2	0	0	**A2**	29	127	1	0	0
	B1	1	13	215	5	6	**B1**	3	10	216	6	5	**B1**	3	7	218	6	6
	B2	0	2	14	18	15	**B2**	0	0	14	28	7	**B2**	0	0	6	28	15
	C1	0	1	15	9	35	**C1**	0	2	13	4	41	**C1**	0	0	3	15	42

Tabela 5: Matrizes de confusão das melhores execuções dos modelos com maior taxa de acerto no cenário de validação cruzada em comparação com a linha de base (76,09%): Albertina PT-PT 100M (82,11%) e GPorTuguese-2 (81,60%).

Abordagem	**REQM**	**TA**	**TAA**	**F_1**
Classificação	0,5341±0,0327	81,77±0,44	96,27±0,54	76,17±1,01
Regressão	**0,4940±0,0268**	80,77±0,58	**97,83±0,60**	75,33±1,12
Média *Softmax*	0,5134±0,0132	**82,33±0,26**	96,93±0,54	**76,74±0,60**
Regressão Ordinal	0,5439±0,0146	80,60±1,31	97,16±0,73	74,69±0,99

Tabela 6: Resultados obtidos pelas abordagens de classificação e regressão no cenário de validação cruzada.

pela função *softmax* e a aplicação de uma abordagem de regressão logística ordinal sobre essa distribuição. De forma semelhante ao observado por Aluisio et al. (2010), podemos ver que as diferenças de desempenho entre as abordagens são pequenas, nunca chegando aos dois pontos percentuais, exceto no caso da REQM, já que o modelo de regressão é treinado especificamente para minimizar essa métrica. Além disso, no nosso estudo original com dez execuções de treino (Ribeiro et al., 2024b), as diferenças foram ainda menores. Isto sugere que o modelo pré-treinado usado como base e os dados utilizados para o ajuste são mais relevantes do que capturar a natureza ordinal dos níveis de inteligibilidade. Ainda assim, é possível que as diferenças se possam tornar mais evidentes se forem considerados dados mais diversificados.

Comparando os resultados em termos de métricas específicas, como referido anteriormente, a abordagem de regressão apresenta uma REQM menor do que a abordagem de classificação. No entanto, isto ocorre às custas de um impacto de cerca de um ponto percentual ao nível da taxa de acerto e da medida F_1, o que sugere que o maior número de neurónios na camada de saída melhora a capacidade do modelo para capturar as características específicas de cada nível, o que pode ser utilizado para obter uma aproximação melhor do nível real de um texto. Por outro lado, o modelo de regressão tem o melhor desempenho quanto à taxa de acerto adjacente. Tal sugere que os erros do modelo de classificação são mais dispersos.

Analisando os resultados das adaptações do modelo de classificação para tentar capturar a relação ordinal entre os níveis, podemos ver que a aplicação da abordagem de regressão ordinal leva a uma melhoria do desempenho ao nível da taxa de acerto adjacente. No entanto, essa melhoria implica um impacto significativo em todas as outras métricas. Além disso, esta abordagem requer dois passos de treino, o que implica um consumo adicional de recursos em comparação com as restantes abordagens. Como tal, a sua aplicação não é vantajosa. Por outro lado, a adaptação baseada na média ponderada da distribuição de probabilidade não só aproxima o desempenho do modelo de classificação ao desempenho do modelo de regressão quanto aos valores de REQM e da taxa de acerto adjacente, como ainda melhora o seu desempenho quanto à taxa de acerto e da medida F_1. Isto sugere que ponderar a probabilidade atribuída a cada nível em vez de simplesmente selecionar aquele com a maior probabilidade é uma abordagem adequada para considerar a natureza ordinal dos níveis.

Para podermos fazer uma análise mais aprofundada do desempenho das diferentes abordagens, a Tabela 7 mostra as matrizes de confusão das execuções com maior taxa de acerto das abordagens de classificação e regressão e da adaptação baseada na média ponderada da distribuição de probabilidade das classes. Tal como observado na comparação entre as versões ajustadas de vários modelos pré-treinados, o nível B1 é aquele para o qual a sensibilidade é maior, o que pode sugerir um certo viés para a sua previsão. No entanto, este é também um dos níveis com maior precisão,

Classificação

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	73	16	3	0	0
	A2	22	133	2	0	0
	B1	3	10	216	6	5
	B2	0	0	14	28	7
	C1	0	2	13	4	41

Regressão

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	72	19	1	0	0
	A2	28	125	4	0	0
	B1	0	9	220	11	0
	B2	0	0	16	31	2
	C1	0	0	11	12	37

Média *Softmax*

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	66	22	3	1	0
	A2	18	137	2	0	0
	B1	0	7	223	3	7
	B2	0	1	11	26	11
	C1	0	2	8	8	42

Tabela 7: Matrizes de confusão dos modelos de classificação e regressão com maior taxa de acerto no cenário de validação cruzada: classificação (82,11%), regressão (81,10%) e média *softmax* (82,61%).

superado apenas pelo nível C1 na abordagem de regressão. Do mesmo modo, mantêm-se as dificuldades em distinguir entre si os níveis A. Nesse sentido, enquanto a abordagem de regressão tem mais tendência em prever exemplos do nível A2 como sendo do nível A1, a adaptação do modelo de classificação tem a tendência inversa. Além disso, considerando os níveis mais avançados, podemos ver que a abordagem de regressão parece ter um viés para os níveis B, enquanto a abordagem de classificação e a sua adaptação têm alguma tendência em sobre-estimar a dificuldade de alguns textos desses níveis. Ainda assim, um problema comum a todas as abordagens é a classificação de um conjunto de textos de nível C1 como sendo do nível B1. Tendo em conta que o nível dos exemplos de treino é inferido a partir do nível do respetivo exame, esta divergência na classificação pode significar que esses textos não são realmente de nível C1, sendo a dificuldade introduzida pelo exercício a realizar.

6.2. Generalização

A Tabela 8 mostra os resultados obtidos quando os modelos treinados no cenário de validação cruzada são aplicados no conjunto de teste. Podemos ver que, em geral, o desempenho dos modelos é superior quando é considerado o nível do exame em vez da anotação da complexidade dos textos feita por peritos. No entanto, isto era expectável, uma vez que os modelos foram treinados em exemplos cujo nível foi aproximado com base no nível do exame. Por outro lado, mesmo considerando o nível do exame, o melhor desempenho é de apenas apenas de 46,88% em taxa de acerto e de 58,39% na medida F_1, o que revela uma fraca capacidade de generalização. Se considerarmos execuções individuais, a abordagem de regressão demonstra uma maior capacidade de generalização, tal como observado por Heilman et al. (2008). Ainda assim, a taxa de acerto máxima continua a ser de apenas 50%. Logo, é importante analisar os resultados com mais detalhe para perceber a causa desta quebra de desempenho em comparação com o cenário de validação cruzada e encontrar soluções para a mitigar.

A Tabela 9 apresenta as matrizes de confusão das melhores execuções das várias abordagens quando aplicadas ao conjunto de teste, considerando o nível do exame. Podemos ver que as principais diferenças entre as abordagens que foram observadas no cenário de validação cruzada também podem ser observadas neste caso. Contudo, também podemos ver que todos os modelos preveem vários exemplos dos níveis A como sendo do nível B1. Sem mais informações, poderíamos assumir que os modelos estão enviesados para prever o nível predominante nos dados de treino. Porém, ao inspecionar esses exemplos, descobrimos que correspondem aos textos curtos, como os mostrados na Tabela 3, que são exclusivos dos exames modelo dos níveis A. O desempenho ligeiramente superior da abordagem de regressão deve-se ao facto de ser a única capaz de classificar corretamente alguns destes textos. A classificação da maioria destes textos como B1 pode ser explicada pelo facto de os textos mais curtos nos dados de treino pertencerem a esse nível, embora sejam significativamente mais longos. Como tal, acreditamos que a incapacidade dos modelos para generalizar o seu desempenho para este tipo de texto pode ser superada pela inclusão de tipos de texto mais diversificados nos dados de treino.

A Tabela 10 mostra os resultados se os textos curtos problemáticos não forem considerados. Neste caso, já é possível observar um desempenho bastante mais próximo, apesar de ainda inferior, ao observado no cenário de validação cruzada. A exceção é a abordagem baseada em características usada como linha de base, que continua a revelar dificuldades de generalização. Isto deve-se a um sobreajustamento aos dados de treino, que potencialmente poderá ser mitigado através do uso de apenas um subconjunto das características. No entanto, deixamos esse estudo para trabalho futuro.

Tal como no cenário de validação cruzada, neste caso, a abordagem de regressão é a que leva à maior taxa de acerto adjacente, atingindo um resultado perfeito. Por outro lado, a abordagem de classificação é a com melhor

	Exame			Anotadores		
Abordagem	**TA**	**TAA**	**F_1**	**TA**	**TAA**	**F_1**
Linha de Base	34,38±0,00	75,00±0,00	39,30±0,00	25,00±0,00	81,25±0,00	29,77±0,00
Classificação	**46,88±0,00**	78,12±0,00	**58,39±0,00**	28,12±0,00	**84,38±0,00**	34,80±0,00
Regressão	**46,88±3,12**	**86,46±1,80**	52,40±2,66	**46,88±3,12**	**84,38±0,00**	**52,62±1,74**
Média *Softmax*	45,83±1,80	78,12±0,00	56,16±3,85	29,17±1,80	**84,38±0,00**	36,76±3,39

Tabela 8: Resultados obtidos no conjunto de teste pelas diferentes abordagens baseadas no ajuste de modelos, considerando os níveis dos exames e as anotações dos peritos.

Linha de Base

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	2	1	5	0	0
	A2	2	1	8	0	1
	B1	1	0	4	0	0
	B2	0	0	0	2	1
	C1	0	0	1	1	2

Classificação

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	3	0	5	0	0
	A2	2	2	8	0	0
	B1	1	0	4	0	0
	B2	0	0	0	3	0
	C1	0	0	1	0	3

Regressão

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	3	1	4	0	0
	A2	1	4	7	0	0
	B1	0	1	4	0	0
	B2	0	0	0	3	0
	C1	0	0	0	2	2

Média *Softmax*

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	3	0	5	0	0
	A2	2	2	8	0	0
	B1	1	0	4	0	0
	B2	0	0	0	3	0
	C1	0	0	1	0	3

Tabela 9: Matrizes de confusão das execuções de cada abordagem com maior taxa de acerto no conjunto de teste considerando o nível do exame: linha de base (34,38%), classificação (46,88%), regressão (50,00%) e média *softmax* (46,88%).

desempenho em termos de taxa de acerto e medida F_1. Além disso, é também a mais estável, não revelando qualquer variação de desempenho entre execuções. Pelo contrário, o desempenho inferior revelado pela adaptação com base na média da distribuição de probabilidades deve-se ao fraco desempenho numa das execuções. Contudo, no nosso estudo com mais execuções (Ribeiro et al., 2024b), observámos que, em média, é possível obter um desempenho superior usando esta adaptação.

Além da dificuldade em classificar os textos curtos, existem mais alguns erros comuns a todas as abordagens. Por exemplo, existe um diálogo extraído do exame modelo do nível A2 que é classificado como sendo de nível A1 por todos os modelos. Existe também a tendência de classificar alguns textos do exame do nível C1 como sendo dos níveis B. No entanto, é importante relembrar que a classificação do nível de complexidade é uma tarefa subjetiva e difícil, até para humanos (Branco et al., 2014a; Curto, 2014). Na verdade, todos os textos que foram agora referidos, foram classificados pelos peritos de forma semelhante aos modelos. Isto sugere que os modelos têm alguma capacidade de identificar exemplos cujo nível está desfasado do nível do exame que os contém.

Voltando a analisar as Tabelas 8 e 10, mas focando agora nos resultados em que são consideradas as anotações dos peritos, podemos ver que, tal como referido anteriormente, com exceção da abordagem de regressão, o desempenho dos modelos em termos de taxa de acerto e medida F_1 é bastante inferior ao atingido quando é considerado o nível do exame. A abordagem de regressão é uma exceção por duas razões. Primeiro, é a abordagem com maior variação entre execuções, permitindo-lhe ter simultaneamente a execução com melhor desempenho considerando o nível do exame e a execução com melhor desempenho considerando as anotações dos peritos, com uma taxa de acerto de 50% em ambos os casos. Segundo, esta abordagem evita prever os níveis extremos, especialmente o nível C1. Os peritos revelaram um comportamento semelhante. Como tal, mesmo as execuções com menor desempenho aproximam-se mais das anotações dos peritos do que as da abordagem de classificação, que tem mais tendência para prever esses níveis.

	Exame			Anotadores		
Abordagem	**TA**	**TAA**	**F_1**	**TA**	**TAA**	**F_1**
Linha de Base	57,89±0,00	84,21±0,00	56,00±0,00	42,10±0,00	89,47±0,00	40,32±0,00
Classificação	**78,95±0,00**	89,47±0,00	**79,81±0,00**	47,37±0,00	**94,74±0,00**	46,60±0,00
Regressão	75,44±2,48	**100,00±0,00**	74,60±2,58	**71,93±2,48**	**94,74±0,00**	**70,22±2,51**
Média *Softmax*	77,19±2,48	89,47±0,00	77,59±3,14	49,12±2,48	**94,74±0,00**	48,56±2,77

Tabela 10: Resultados obtidos no conjunto de teste pelas diferentes abordagens baseadas no ajuste de modelos quando os textos curtos extraídos dos exames modelo dos níveis A não são considerados.

Linha de Base

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	1	1	4	0	0
	A2	3	1	10	0	0
	B1	1	0	2	0	1
	B2	0	0	2	3	2
	C1	0	0	0	0	1

Classificação

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	2	0	4	0	0
	A2	3	1	10	0	0
	B1	1	1	2	0	0
	B2	0	0	2	3	2
	C1	0	0	0	0	1

Regressão

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	2	0	4	0	0
	A2	0	6	8	0	0
	B1	1	1	2	0	0
	B2	0	0	1	5	1
	C1	0	0	0	0	1

Média *Softmax*

		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	2	0	4	0	0
	A2	3	1	10	0	0
	B1	1	1	2	0	0
	B2	0	0	2	4	1
	C1	0	0	0	0	1

Tabela 11: Matrizes de confusão das execuções de cada abordagem com maior taxa de acerto no conjunto de teste considerando as anotações dos peritos: linha de base (25,00%), classificação (28,12%), regressão (50,00%) e média *softmax* (31,25%).

Estas tendências podem ser confirmadas nas matrizes de confusão apresentadas na Tabela 11.

Por fim, na Tabela 12, podemos ver o desempenho dos modelos considerando apenas os textos em que existe concordância entre o nível do exame e a anotação dos peritos. Verifica-se que o desempenho de todas as abordagens neste subconjunto é superior ao observado no conjunto de teste completo, independentemente da anotação considerada. No entanto, grande parte desta melhoria no desempenho se deve ao facto de cerca de metade dos textos curtos não estarem a ser considerados. Ainda assim, é interessante constatar que a abordagem de regressão é aquela com melhor desempenho em todas as métricas, falhando apenas a classificação de alguns dos restantes textos curtos. Por outro lado, além dos textos curtos, a abordagem de classificação e a sua adaptação classificam erradamente, como sendo do nível A1, o exemplo do nível A2 mostrado no Apêndice A.2.

Em geral, considerando a quantidade reduzida de dados disponível para treinar os modelos e a inferência do seu nível de complexidade a partir do nível do exame, a abordagem de regressão parece ser aquela com maior capacidade de generalização, uma vez que tem menos tendência para sobreajustamento. Assim, se se dispusesse de um conjunto de treino maior, mais diversificado e mais representativo, esperar-se-ia que a abordagem de classificação e, em particular, a sua adaptação com base na média ponderada da distribuição de probabilidades, atingisse um desempenho superior.

6.3. Modelos Baseados em Instruções

A Tabela 13 mostra os resultados obtidos usando modelos baseados em instruções num cenário *zero-shot*. Em primeiro lugar, é preciso constatar que é difícil encontrar padrões interessantes nos resultados. Ainda assim, podemos ver que o melhor desempenho ao nível de todas as métricas é obtido por modelos da família GPT. O modelo e instrução que levam ao melhor desempenho variam consoante a métrica e se é considerado o nível do exame ou a anotação dos peritos.

Considerando o nível do exame, o melhor desempenho em termos de taxa de acerto (55,21%) e medida F_1 (56,63%) é atingido pelo modelo

Abordagem	**TA**	**TAA**	**F_1**
Linha de Base	40,00±0,00	**93,33±0,00**	41,78±0,00
Classificação	55,33±0,00	**93,33±0,00**	63,00±0,00
Regressão	**64,44±3,14**	**93,33±0,00**	**72,89±1,89**
Média *Softmax*	55,33±0,00	**93,33±0,00**	63,00±0,00

Tabela 12: Resultados obtidos no conjunto de teste pelas diferentes abordagens baseadas no ajuste de modelos, considerando apenas os textos em que a anotação dos peritos coincide com o nível do exame.

		Exame			**Anotadores**		
Modelo	**I**	**TA**	**TAA**	**F_1**	**TA**	**TAA**	**F_1**
GPT 3.5 Turbo	A	**55,21±1,47**	92,71±1,47	**56,63±0,96**	37,50±2,55	95,83±1,47	39,97±2,93
	B	44,79±5,31	89,58±1,47	41,21±6,05	22,92±2,95	89,58±1,47	23,38±3,66
GPT 4 Turbo	A	30,21±1,47	**93,75±0,00**	28,19±1,78	**53,12±0,00**	95,83±1,47	40,35±0,11
	B	36,46±1,47	91,67±1,47	34,21±3,76	50,00±2,55	94,79±1,47	**48,70±8,94**
GPT 4o	A	44,79±1,47	92,71±1,47	47,26±1,28	38,54±1,47	96,88±0,00	38,82±0,96
	B	34,38±0,00	90,62±0,00	33,84±1,37	45,83±2,95	93,75±0,00	41,56±2,87
GPT 4o Mini	A	34,38±0,00	90,62±0,00	37,22±0,00	43,75±0,00	**100,00±0,00**	43,27±0,00
	B	37,50±0,00	90,62±0,00	42,54±0,00	46,88±0,00	**100,00±0,00**	47,58±0,00
Llama 3 8B	A	31,25±0,00	87,50±0,00	26,78±0,00	43,75±0,00	87,50±0,00	34,67±0,00
	B	34,38±0,00	87,50±0,00	26,91±0,00	50,00±0,00	90,62±0,00	38,94±0,00
Mixtral 8x7B	A	43,75±0,00	84,38±0,00	29,52±0,00	31,25±0,00	96,88±0,00	24,24±0,00
	B	46,88±0,00	81,25±0,00	31,75±0,00	31,25±0,00	90,62±0,00	21,50±0,00

Tabela 13: Resultados obtidos no conjunto de teste usando modelos baseados em instruções. A coluna I distingue entre as duas instruções usadas no nosso estudo.

GPT 3.5 Turbo, o mais antigo de todos os explorados, quando emparelhado com a instrução A, que foca diretamente no nível de complexidade do texto. Por outro lado, considerando a anotação dos peritos, o melhor desempenho é atingido pelo modelo GPT 4 Turbo. No entanto, enquanto a maior taxa de acerto (53,12%) é atingida usando a instrução A, a maior medida F_1 (48,70%) é atingida usando a instrução B, que foca no nível de proficiência necessário para compreender o texto. Já a maior taxa de acerto adjacente é atingida pelo modelo GPT 4 Turbo quando é considerado o nível do exame e pelo modelo GPT 4o Mini quando são consideradas as anotações dos peritos. Contudo, em ambos os casos, este desempenho surge às custas de uma quebra elevada em termos das restantes métricas, especialmente quando considerando o nível do exame.

Considerando os resultados referidos anteriormente, poderíamos ser levados a assumir que os modelos baseados em instruções têm um desempenho melhor quando é considerado o nível do exame, pois podem potencialmente ter visto os exames modelo durante o seu treino. Porém, analisando a Tabela 13, podemos ver que o modelo GPT 3.5 Turbo é uma exceção e que, na maioria dos casos, o desempenho é melhor quando são consideradas as anotações dos peritos.

Em comparação com os resultados dos modelos de base ajustados para a tarefa, o desempenho dos melhores modelos baseados em instruções é superior em termos da taxa de acerto e da taxa de acerto adjacente, mas inferior ao nível da medida F_1. Analisando as matrizes de confusão na Tabela 14, podemos ver que o aumento da taxa de acerto se deve ao facto de os modelos não estarem enviesados para classificar os textos curtos como sendo do nível B1. Por outro lado, a medida F_1 inferior deve-se a uma maior confusão entre os níveis A e à incapacidade de o modelo GPT 4 Turbo prever o nível C1. Analisando a confusão entre os níveis A com mais detalhe, descobrimos que esta está, mais uma vez, relacionada com os textos curtos, já que os modelos baseados em instruções têm tendência a classificar a maioria desses textos como sendo do nível A1.

Relativamente aos modelos disponíveis publicamente, podemos ver que, na maioria dos casos, atingem melhores resultados quando emparelhados com a instrução B. Além disso, enquanto o modelo da família Llama tem melhor desempenho quando são consideradas as anotações dos peritos, o modelo da família Mistral tem melhor desempenho quando é considerado o nível do exame. Em ambos os casos, estes modelos ficam em segundo lugar quanto à taxa de acerto e, ainda assim, distantes do melhor desempenho.

		Exame				
		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	5	2	1	0	0
	A2	5	5	1	1	0
	B1	0	0	3	2	0
	B2	0	0	0	2	1
	C1	0	0	0	1	3

		Anotadores				
		Previsto				
		A1	**A2**	**B1**	**B2**	**C1**
Real	**A1**	4	2	0	0	0
	A2	4	6	3	1	0
	B1	0	1	1	2	0
	B2	0	0	1	6	0
	C1	0	0	0	1	0

Tabela 14: Matrizes de confusão das melhores execuções dos modelos baseados em instruções com maior taxa de acerto no conjunto de teste, considerando os níveis dos exames (GPT 3.5 Turbo, Instrução A, 56,25%) e as anotações dos peritos (GPT 4 Turbo, Instrução A, 53,12%).

Alguns comentários mais genéricos sobre os modelos: ao contrário dos modelos disponíveis publicamente, mesmo otimizando os hiperparâmetros para esse efeito, é impossível garantir a reprodutibilidade dos resultados dos modelos da família GPT; os modelos das famílias GPT e Mistral têm alguma dificuldade em seguir a instrução de responder apenas com o nível, independentemente da forma como essa instrução é dada; o modelo da família Mistral inclui dois níveis na resposta em múltiplas ocasiões.

Tendo em conta o desempenho do modelo GPT 3.5 Turbo em conjunto com a instrução A no conjunto de teste quando considerado o nível do exame, decidimos aplicá-lo também ao conjunto de treino, de forma a comparar os resultados com o do cenário de validação cruzada. Contudo, neste caso, a taxa de acerto média foi de apenas 38,18% e a medida F_1 de apenas 33,94%.

Em geral, apesar de aparentarem ter menos enviesamentos que os modelos ajustados especificamente para a tarefa e de serem capazes de lidar com tipos de texto mais diversos, a maioria dos resultados obtidos neste estudo sugerem que os modelos baseados em instruções não têm, de base, capacidade para identificar adequadamente o nível de complexidade dos textos. No entanto, é possível que desenvolvam essa capacidade se as instruções incluírem exemplos adequados. Como tal, no futuro, é importante explorar o desempenho destes modelos num cenário *few-shot*.

7. Conclusão

Neste estudo, explorámos a avaliação automática do nível de complexidade de textos em português europeu. Para isso, começámos por complementar o conjunto de textos extraídos de exames para avaliação do nível de proficiência usado em estudos anteriores com um novo conjunto de teste disponível publicamente e anotado não só por inferência a partir do nível do exame, como também por peritos. Em seguida, analisámos o desempenho de múltiplos modelos de base pré-treinados quando ajustados para desempenhar esta tarefa. Além disso, explorámos não só o uso de abordagens de classificação, como também de regressão, numa tentativa de ter em conta a natureza ordinal dos níveis de complexidade definidos pelo CEFR. Por fim, fizemos uma análise preliminar da capacidade de modelos baseados em instruções desempenharem esta tarefa num cenário *zero-shot* (sem exemplos).

As nossas experiências num cenário de validação cruzada mostraram que, considerando a quantidade reduzida de dados de treino, o melhor desempenho pode ser atingido através do ajuste da versão mais pequena do modelo Albertina PT-PT. Além disso, uma adaptação do modelo de classificação, baseada na previsão com base na média ponderada da distribuição de probabilidades em vez da simples seleção da classe mais provável, leva a previsões mais robustas e, em geral, a um melhor desempenho.

Contudo, também devido à escassez de dados de treino, estes modelos não generalizam bem para diferentes tipos de texto nem para as anotações dos peritos, ficando sobreajustados aos níveis inferidos a partir do nível do exame. Um modelo baseado em regressão pura tem menos tendência a ficar sobreajustado e, por isso, generaliza melhor para tipos de texto não observados durante o treino. No entanto, em contrapartida, o desempenho global é mais baixo. Como tal, tal como para muitas outras tarefas de PLN em línguas com poucos recursos, é importante obter um conjunto de dados de treino maior, mais diversificado e mais representativo, de forma a treinar melhores modelos.

Relativamente ao uso de modelos baseados em instruções, apesar de aparentarem ter menos enviesamentos do que os modelos ajustados especificamente para a tarefa e de serem capazes de lidar com tipos de texto mais diversos, demonstraram também um comportamento bastante errático e de difícil análise, sendo o melhor desempenho em relação aos modelos ajustados atingido apenas em casos muito específicos.

Como tal, argumentamos que a maioria destes modelos não tem, de base, capacidade para identificar adequadamente o nível de complexidade dos textos. Contudo, é possível que venham a desenvolver essa capacidade se as instruções incluírem exemplos adequados. Como tal, como trabalho futuro, pretendemos explorar o desempenho destes modelos em cenários *few-shot*, usando diferentes abordagens para selecionar os exemplos.

Caso a abordagem *few-shot* não resulte, para mitigar os problemas criados pela escassez de dados anotados, pretendemos explorar o uso de textos na variedade brasileira da língua durante o treino e averiguar se as vantagens do volume de dados de treino adicional superam os problemas introduzidos pelas diferenças entre as duas variedades da língua. Esta abordagem pode também ser generalizada para o uso de dados anotados noutras línguas em combinação com modelos de base multilingues.

Uma outra abordagem a explorar no futuro é o uso de modelos híbridos que combinam características linguísticas com modelos de língua, que já revelaram ser capazes de atingir um desempenho superior à soma das partes em estudos anteriores sobre a classificação automática do nível de complexidade de textos (Correia & Mendes, 2021; Wilkens et al., 2024).

Por fim, considerando a subjetividade da avaliação do nível de inteligibilidade ou de complexidade de textos e as suas potenciais aplicações, é importante fazer um esforço para o desenvolvimento de modelos interpretáveis para esta tarefa, de modo a entender porque é que um texto é classificado como sendo de um determinado nível e como ele pode ser alterado para ser mais adequado ao nível de proficiência do público-alvo.

Agradecimentos

Este trabalho foi financiado por fundos nacionais portugueses através da Fundação para a Ciência e a Tecnologia (FCT) (Referência: UIDB/50021/2020, DOI: 10.54499/UIDB/50021/2020) e pela Comissão Europeia (Projeto: iRead4Skills, Número da subvenção: 1010094837, Tópico: HORIZON-CL2-2022-TRANSFORMATIONS-01-07, DOI: 10.3030/101094837).

Gostaríamos de agradecer ao Camões, I.P., por nos ter dado acesso aos textos usados nos seus exames e a permissão para os usar para treinar e avaliar os nossos modelos. Gostaríamos também de agradecer aos peritos que se disponibilizaram para anotar os textos do conjunto de teste.

Referências

Akef, Soroosh, Amália Mendes, Detmar Meurers & Patrick Rebuschat. 2024. Investigating the generalizability of Portuguese readability assessment models trained using linguistic complexity features. Em *International Conference on Computational Processing of Portuguese (PROPOR)*, 332–341.

Aluisio, Sandra, Lucia Specia, Caroline Gasperin & Carolina Scarton. 2010. Readability assessment for text simplification. Em *NAACL-HLT Workshop on Innovative Use of NLP for Building Educational Applications*, 1–9.

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson et al. 2021. On the opportunities and risks of foundation models. arXiv [cs.LG/cs.AI/cs.CY]. 10.48550/arXiv.2108.07258

Branco, António, João Rodrigues, Francisco Costa, João Silva & Rui Vaz. 2014a. Assessing automatic text classification for interactive language learning. Em *International Conference on Information Society (i-Society)*, 70–78. 10.1109/i-society.2014.7009014

Branco, António, João Rodrigues, Francisco Costa, João Silva & Rui Vaz. 2014b. Rolling out text categorization for language learning assessment supported by language technology. Em *International Conference on the Computational Processing of the Portuguese Language (PROPOR)*, 256–261. 10.1007/978-3-319-09761-9_29

Cha, Miriam, Youngjune Gwon & H.T. Kung. 2017. Language modeling by clustering with word embeddings for text readability assessment. Em *Conference on Information and Knowledge Management (CIKM)*, 2003–2006. 10.1145/3132847.3133104

Chen, Banghao, Zhaofeng Zhang, Nicolas Langrené & Shengxin Zhu. 2023. Unleashing the potential of prompt enginee-

ring in large language models: a comprehensive review. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2310.14735

Correia, João & Rui Mendes. 2021. Neural complexity assessment: A deep learning approach to readability classification for European Portuguese corpora. Em *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, 300–311. 10.1007/978-3-030-91608-4_30

Council of Europe. 2001. *Common european framework of reference for languages: Learning, teaching, assessment.* Cambridge University Press.

Crossley, Scott A., Stephen Skalicky, Mihai Dascalu, Danielle S. McNamara & Kristopher Kyle. 2017. Predicting text comprehension, processing, and familiarity in adult readers: New approaches to readability formulas. *Discourse Processes* 54(5–6). 340–359. 10.1080/0163853x.2017.1296264

Curto, Pedro. 2014. *Classificador de Textos para o Ensino de Português como Segunda Língua*: Instituto Superior Técnico, Universidade de Lisboa. Tese de Mestrado.

Curto, Pedro, Nuno Mamede & Jorge Baptista. 2015. Automatic text difficulty classifier. Em *International Conference on Computer Supported Education (CSEDU)*, 36–44. 10.5220/0005428300360044

Devlin, Jacob, Ming-Wei Chang, Lee Kenton & Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 4171–4186. 10.18653/v1/N19-1423

Direção de Serviços de Língua e Cultura. 2017. *Referencial Camões PLE.* Camões, Instituto da Cooperação e da Língua I.P.

DuBay, William H. 2004. *The principles of readability.* Impact Information.

Filighera, Anna, Tim Steuer & Christoph Rensing. 2019. Automatic text difficulty estimation using embeddings and neural networks. Em *European Conference on Technology Enhanced Learning (EC-TEL)*, 335–348. 10.1007/978-3-030-29736-7_25

Forti, Luciana, Giuliana Grego Bolli, Filippo Santarelli, Valentino Santucci & Stefania Spina. 2020. MALT-IT2: a new resource to measure text difficulty in light of CEFR levels for Italian L2 learning. Em *Language Resources and Evaluation Conference (LREC)*, 7204–7211.

François, Thomas & Cédrick Fairon. 2012. An "AI readability" formula for French as a foreign language. Em *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 466–477.

François, Thomas, Adeline Müller, Eva Rolin & Magali Norré. 2020. AMesure: A web platform to assist the clear writing of administrative texts. Em *Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (AACL-IJCNLP)*, 1–7. 10.18653/v1/2020.aacl-demo.1

Friedman, Jerome, Trevor Hastie & Robert Tibshirani. 2000. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics* 28(2). 337–407. 10.1214/aos/1016218223

Friedman, Jerome H. 2001. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* 29(5). 10.1214/aos/1013203451

Giray, Louie. 2023. Prompt engineering with ChatGPT: a guide for academic writers. *Annals of Biomedical Engineering* 51(12). 2629–2633. 10.1007/s10439-023-03272-4

Grosso, Maria José, António Soares, Fernanda de Sousa & José Pascoal. 2011. QuaREPE: Quadro de referência para o ensino Português no estrangeiro – documento orientador. Relatório técnico. Direção-Geral da Educação (DGE).

Guillou, Pierre. 2020. Faster than training from scratch: Fine-tuning the English GPT-2 in any language with Hugging Face and FastAI v2 (practical case with Portuguese). Medium.

Hartmann, Nathan, Erick Fonseca, Christopher Shulby, Marcos Treviso, Jéssica Silva & Sandra Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. Em *Brazilian Symposium in Information and Human Language Technology (STIL)*, 122–131.

He, Pengcheng, Xiaodong Liu, Jianfeng Gao & Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. Em *International Conference on Learning Representations (ICLR)*,

Heilman, Michael, Kevyn Collins-Thompson & Maxine Eskenazi. 2008. An analysis of statistical models and features for reading difficulty prediction. Em *Workshop on Innovative use of NLP for Building Educational Applications (BEA)*, 71–79.

Hernandez, Nicolas, Nabil Oulbaz & Tristan Faine. 2022. Open corpora and toolkit for assessing text readability in French. Em *Proceedings of the Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI)*, 54–61.

Hulstijn, Jan H. 2007. The shaky ground beneath the CEFR: Quantitative and qualitative dimensions of language proficiency. *The Modern Language Journal* 91(4). 663–667. 10.1111/j.1540-4781.2007.00627_5.x

Hulstijn, Jan H. 2011. Language proficiency in native and nonnative speakers: An agenda for research and suggestions for second-language assessment. *Language Assessment Quarterly* 8(3). 229–249. 10.1080/15434303.2011.565844

Jiang, Albert Q., Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix & William El Sayed. 2023. Mistral 7B. arXiv [cs.CL/cs.AI/cs.LG]. 10.48550/arXiv.2310.06825

Jiang, Albert Q., Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix & William El Sayed. 2024. Mixtral of experts. arXiv [cs.LG/cs.CS]. 10.48550/arXiv.2401.04088

Jönsson, Simon, Evelina Rennes, Johan Falkenjack & Arne Jönsson. 2018. A component based approach to measuring text complexity. Em *7th Swedish Language Technology Conference (SLTC)*, 58–61.

Karpov, Nikolay, Julia Baranova & Fedor Vitugin. 2014. Single-sentence readability prediction in Russian. Em *International Conference on Analysis of Images, Social Networks and Texts (AIST)*, 91–100. 10.1007/978-3-319-12580-0_9

Kincaid, J. Peter, Robert P. Fishburne Jr, Richard L. Rogers & Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Relatório técnico. Institute for Simulation and Training, University of Central Florida. 10.21236/ada006655

Leal, Sidney Evaldo & Sandra Maria Aluísio. 2024. Complexidade textual e suas tarefas relacionadas. Em Helena. M. Caseli & Maria G. V. Nunes (eds.), *Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português*, chap. 23. BPLN 2nd edn.

Leal, Sidney Evaldo, Magali Sanches Duran, Carolina Evaristo Scarton, Nathan Siegle Hartmann & Sandra Maria Aluísio. 2023. NILC-Metrix: Assessing the complexity of written and spoken language in Brazilian Portuguese. *Language Resources and Evaluation* 73–110. 10.1007/s10579-023-09693-w

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer & Veselin Stoyanov. 2019. RoBERTa: a robustly optimized BERT pretraining approach. arXiv [cs.CL]. 10.48550/arXiv.1907.11692

Llama Team. 2024. The Llama 3 herd of models. arXiv [cs.AI/cs.CL/cs.CV]. 10.48550/arXiv.2407.21783

Martinc, Matej, Senja Pollak & Marko Robnik-Šikonja. 2021. Supervised and unsupervised neural approaches to text readability. *Computational Linguistics* 47(1). 141–179. 10.1162/coli_a_00398

Marujo, Luís, José Lopes, Nuno Mamede, Isabel Trancoso, Juan Pino, Maxine Eskenazi, Jorge Baptista & Céu Viana. 2009. Porting REAP to European Portuguese. Em *International Workshop on Speech and Language Technology in Education (SLaTE)*, 69–72.

McCullagh, Peter. 1980. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B (Methodological)* 42(2). 109–127. 10.1111/j.2517-6161.1980.tb01109.x

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado & Jeff Dean. 2013. Distributed representations of words and phrases and

their compositionality. Em *Advances in Neural Information Processing Systems (NIPS)*, 3111–3119.

Mishra, Swaroop, Daniel Khashabi, Chitta Baral & Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. Em *Annual Meeting of the Association for Computational Linguistics (ACL)*, 3470–3487. 10.18653/v1/2022.acl-long.244

Mohtaj, Salar, Babak Naderi & Sebastian Möller. 2022. Overview of the GermEval 2022 shared task on text complexity assessment of German text. Em *GermEval Workshop on Text Complexity Assessment of German Text*, 1–9.

Nadeem, Farah & Mari Ostendorf. 2018. Estimating linguistic complexity for science texts. Em *Workshop on Innovative Use of NLP for Building Educational Applications*, 45–55. 10.18653/v1/W18-0505

North, Kai, Marcos Zampieri & Matthew Shardlow. 2023. Lexical complexity prediction: An overview. *ACM Computing Surveys* 55(9). 1–42. 10.1145/3557885

OpenAI. 2023. ChatGPT. Online.

OpenAI. 2024. GPT-4 technical report. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2303.08774

Ouyang, Long, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike & Ryan Lowe. 2022. Training language models to follow instructions with human feedback. Em *Advances in Neural Information Processing Systems (NIPS)*, 27730–27744.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot & Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12(84). 2825–2830.

Pilán, Ildikó & Elena Volodina. 2018. Investigating the importance of linguistic complexity features across different datasets related to language learning. Em *Workshop on Linguistic Complexity and Natural Language Processing*, 49–58.

Radford, Alec, Karthik Narasimhan, Tim Salimans & Ilya Sutskever. 2018. Improving language understanding by generative pre-training. OpenAI Blog.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei & Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI Blog.

Reynolds, Robert. 2016. Insights from Russian second language readability classification: Complexity-dependent training requirements, and feature evaluation of multiple categories. Em *Workshop on Innovative Use of NLP for Building Educational Applications*, 289–300. 10.18653/v1/W16-0534

Ribeiro, Eugénio, Nuno Mamede & Jorge Baptista. 2024a. Automatic text readability assessment in European Portuguese. Em *International Conference on Computational Processing of Portuguese (PROPOR)*, 97–107.

Ribeiro, Eugénio, Nuno Mamede & Jorge Baptista. 2024b. Text readability assessment in European Portuguese: A comparison of classification and regression approaches. Em *International Conference on Computational Processing of Portuguese (PROPOR)*, 551–557.

Rodrigues, João, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso & Tomás Osório. 2023. Advancing neural encoding of Portuguese with transformer Albertina PT-*. Em *Portuguese Conference on Artificial Intelligence (EPIA)*, 441–453. 10.1007/978-3-031-49008-8_35

Sanh, Victor, Lysandre Debut, Julien Chaumond & Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. arXiv [cs.CL]. 10.48550/arXiv.1910.01108

Santos, Rodrigo, João Rodrigues, António Branco & Rui Vaz. 2021. Neural text categorization with transformers for learning Portuguese as a second language. Em *Portuguese Conference on Artificial Intelligence (EPIA)*, 715–726. 10.1007/978-3-030-86230-5_56

Santos, Rodrigo, João Rodrigues, Luís Gomes, João Ricardo Silva, António Branco, Henrique Lopes Cardoso, Tomás Freitas Osório & Bernardo Leite. 2024. Fostering the Ecosystem of Open Neural Encoders for Portuguese with Albertina PT* Family. Em *Proceedings of the Annual Meeting of the Special Interest Group on Under-resourced Languages (SIGUL)*, 105–114.

Santucci, Valentino, Filippo Santarelli, Luciana Forti & Stefania Spina. 2020. Automatic classification of text complexity. *Applied Sciences* 10(20). 7285. 10.3390/app10207285

Scarton, Carolina Evaristo & Sandra Maria Aluísio. 2010. Análise da inteligibilidade de textos via ferramentas de processamento de língua natural: Adaptando as métricas do Coh-Metrix para o Português. *Linguamática* 2(1). 45–61.

Seabold, Skipper & Josef Perktold. 2010. Statsmodels: Econometric and statistical modeling with Python. Em *Python in Science Conference (SciPy)*, 92–96. 10.25080/majora-92bf1922-011

Souza, Fábio, Rodrigo Nogueira & Roberto Lotufo. 2020. BERTimbau: Pretrained BERT models for Brazilian Portuguese. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 403–417. 10.1007/978-3-030-61377-8_28

Suárez, Pedro Javier Ortiz, Benoît Sagot & Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. Em *Workshop on the Challenges in the Management of Large Corpora (CMLC)*, 9–16. 10.14618/ids-pub-9021

Sung, Yao Ting, Wei Chun Lin, Scott Benjamin Dyson, Kuo En Chang & Yu Chia Chen. 2015. Leveling L2 texts through readability: Combining multilevel linguistic features with the CEFR. *The Modern Language Journal* 99(2). 371–391. 10.1111/modl.12213

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave & Guillaume Lample. 2023. LLaMA: Open and efficient foundation language models. arXiv [cs.CL]. 10.48550/arXiv.2302.13971

Wagner Filho, Jorge A., Rodrigo Wilkens, Marco Idiart & Aline Villavicencio. 2018. The brWaC corpus: a new open resource for Brazilian Portuguese. Em *International Conference on Language Resources and Evaluation (LREC)*, 4339–4344.

Wilkens, Rodrigo, David Alfter, Xiaoou Wang, Alice Pintard, Anaïs Tack, Kevin Yancey & Thomas François. 2022. FABRA: French aggregator-based readability assessment toolkit. Em *Language Resources and Evaluation Conference (LREC)*, 1217–1233.

Wilkens, Rodrigo, Patrick Watrin, Rémi Cardon, Alice Pintard, Isabelle Gribomont & Thomas François. 2024. Exploring hybrid approaches to readability: Experiments on the complementarity between linguistic features and transformers. Em *Findings of the Association for Computational Linguistics*, 2316–2331.

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest & Alexander M. Rush. 2020. HuggingFace's transformers: State-of-the-art natural language processing. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 38–45. 10.18653/v1/2020.emnlp-demos.6

Xia, Menglin, Ekaterina Kochmar & Ted Briscoe. 2016. Text readability assessment for second language learners. Em *Workshop on Innovative Use of NLP for Building Educational Applications*, 12–22. 10.18653/v1/W16-0502

Yancey, Kevin, Alice Pintard & Thomas Francois. 2021. Investigating readability of french as a foreign language with deep learning and cognitive and pedagogical features. *Lingue e Linguaggio* 20(2). 229–258. 10.1418/102814

Zago, Ricardo. 2023. BERTugues base (aka "bertugues-base-portuguese-cased"). Hugging Face.

A. Exemplos

Este apêndice mostra um exemplo de cada nível de complexidade, extraído dos exames modelo do Camões, I.P., e selecionado de entre os casos em que existe concordância entre o nível do exame e o atribuído pelos anotadores. Quando disponíveis, as referências às fontes dos textos estão incluídas por uma questão de atribuição. No entanto, estas não são consideradas parte do texto.

A.1. A1

Dois irmãos, o Diogo e o Carlos, vão comprar uma prenda para o pai.

Diogo: Carlos, este CD é bom. Gosto muito deste cantor!
Carlos: Sim, eu também gosto. Mas o pai prefere livros, não achas?
Diogo: Sim, tens razão. Ele está sempre a ler.
Carlos: Pois está! Podemos dar-lhe um livro sobre reis e rainhas.
Diogo: Boa ideia. O pai gosta de História de Portugal.
Carlos: Vê lá este aqui, chama-se Filipa de Lencastre.
Diogo: Esse livro deve ser giro. Acho que o pai vai gostar. Quanto é que custa?
Carlos: Custa 15€. Não é muito caro, pois não?
Diogo: Não, é um bom preço.
Carlos: Vamos então levar este livro? Temos aqui esse dinheiro.
Diogo: Sim, acho que é um bom presente.

A.2. A2

Portugal nasceu no Norte. Foi na região Porto e Norte que os portugueses começaram enquanto povo e nação.

O Porto é a grande porta de entrada e pode ser ponto de partida para uma viagem pela diversidade natural e cultural da região. É conhecido pelo vinho e por arquitetos como Álvaro Siza Vieira e Souto de Moura.

O rio Douro atravessa a região. Entra em Portugal apertado por montanhas do interior para percorrer toda a paisagem onde se cultiva o vinho do Porto.

Turismo de Portugal, https://www.visitportugal.com/pt-pt/destinos/porto-e-norte (texto adaptado)

A.3. B1

Dádivas

Anualmente, no Natal, são oferecidos animais. Muitos deles, comprados a lojas. Enquanto isso, nas associações de animais, existem milhares que esperam uma casa quente e o colo de alguém.

A maior parte já foi abandonada pelo mesmo colo que lhe jurou amor eterno, mas que o largou à primeira dificuldade. As desculpas já as conhecemos: ou porque se vai de férias, ou porque nasceu um bebé, ou porque o animal está doente e não se tem dinheiro para tratá-lo.

O problema está em ver-se um animal como um objeto em vez de se pensar nele como um ser que faz parte de uma família. Os animais criam ligações emocionais, têm memória e sofrem quando magoados física e psicologicamente. Se assim não fosse, como explicaríamos os casos de animais que arriscam a vida para salvar os seus companheiros humanos?

Antes de se ter um animal, deve-se refletir se se estará disposto a enfrentar todas as dificuldades. Um animal não é uma prenda. É uma dádiva que só alguns saberão reconhecer.

Ana Bacalhau, *Notícias Magazine*, 28 dezembro 2014 (texto adaptado)

A.4. B2

Imitar o voo das aves

Há quem veja os voadores como "doidos" do ar, por sentirem um desejo irreprimível de voar. O mais curioso é que esse desejo sempre está presente na humanidade, provavelmente desde o dia em que se reparou no voo dos pássaros. Aquilo que terá começado como uma mera contemplação, transformou-se numa obsessão: imitar as aves.

Embora voar fosse para a maioria das pessoas um sonho impossível, algo que estaria para além da capacidade humana, vários foram os que acreditaram e tudo fizeram para o ver concretizado. Uns ficaram-se por cálculos e arabescos em folhas de papel, outros, mais audazes, lançaram-se no vazio com as engenhocas voadoras que construíram. Desses, muitos pagaram com a própria vida a ousadia, outros colecionaram desastres e ferimentos graves. Afortunadamente, um punhado de visionários conseguiu ser bem-sucedido, inscrevendo para sempre os seus nomes na história da aviação. [...]

Da Grécia Antiga, chega-nos o relato daquela que poderá ter sido a primeira máquina voadora movimentada por meios próprios. Diz-se que foi construída por volta de 400 a.C., pelo ilustre matemático Arquitas de Tarento, e seria um pombo de madeira movido a jatos de vapor, que dizem ter voado cerca de duas centenas de metros.

A poesia árabe exalta os feitos de Abbas ibn Firnas (810–887), que alguns consideram ter sido a primeira pessoa a construir um engenho voador e a voar. [...]

O inventor e artista Leonardo da Vinci, génio do Renascimento, passou anos a decifrar o voo das aves e a conceber planos meticulosos para máquinas voadoras. Porém, como não conseguiu perceber as leis da física associadas ao voo, acabou por deixar o mundo dos vivos sem nunca ter construído as suas máquinas ou experimentado o prazer de voar.

Revista *Super Interessante*, novembro de 2012 (texto adaptado e com supressões)

A.5. C1

A mentira da criatividade

O sentido da palavra criatividade banalizou-se. Hoje não existe anúncio de emprego para empresas ou instituições que não tenha um item a proclamar que se privilegiam pessoas criativas, com capacidade inovadora e ideias fora da caixa.

E se o mercado quer, é sintomático que surjam cada vez mais pessoas com esse perfil. É natural que assim seja. Não existe político ou empresário que nos últimos anos não debite com aparente convicção que inovar é necessário.

Ontem um letreiro avisava-me que tinha acabado de entrar numa 'pastelaria criativa'. Ainda olhei para o meu palmier simples a ver se tinha linhas complexas desenhadas por um designer ou se o sabor era exótico, mas não, era apenas um simples, honesto e saboroso palmier. E suspirei de alívio.

Não me interpretem mal. Ser criativo, inovador e ter ideias fora da caixa, é óptimo. Mas não existem muitas pessoas com esse perfil. E tenho sérias dúvidas que as empresas as desejem. O que temos hoje em dia é a romantização dessas noções e sua apropriação superficial, através da retórica que as envolve.

Por um lado vemos cada vez mais difundido o desígnio de que todos podemos triunfar com uma boa ideia. Na verdade celebramos aqueles que consideramos serem criativos, mas apenas a partir do resultado que produziram. A realidade é esta: a maior parte das pessoas, empresas ou instituições tem dificuldade em lidar com indivíduos realmente criativos.

A prosa poética e o amor pelas histórias vieram da mãe: "Era professora primária e contava muito bem histórias de raiz popular. Mas também histórias de fadas. Eu adorava."

Há exceções? Há, como em tudo. Mas não passam disso.

Vítor Belanciano, in *Público*, 26/04/2015 (texto com supressões)

Proposta recebida em Setembro 2024 e aceite para publicação em Novembro 2024.

Reconhecimento de Entidades Nomeadas e Vazamento de Dados em Textos Legislativos: Uma Reavaliação da Literatura

Named Entity Recognition and Data Leakage in Legislative Texts: A Literature Reassessment

Rafael Oleques Nunes
Universidade Federal do Rio Grande do Sul

André Susliz Spritzer
Universidade Federal do Rio Grande do Sul

Carla Maria Dal Sasso Freitas
Universidade Federal do Rio Grande do Sul

Dennis Giovani Balreira
Universidade Federal do Rio Grande do Sul

Resumo

Este trabalho trata do vazamento de dados no treinamento de modelos de Reconhecimento de Entidades Nomeadas (NER) em textos legislativos em português brasileiro, resultante de duplicatas e anotações inconsistentes, o que compromete a avaliação dos modelos. Após corrigir esse vazamento no *corpus* UlyssesNER-Br, foi realizado um novo *benchmark*, comparando os resultados com estudos anteriores em um cenário mais confiável. Também foi reavaliada uma abordagem semissupervisionada utilizando autoaprendizado e amostragem ativa. No entanto, ao reutilizar um *threshold* fixo, escolhido a partir de uma nuvem de valores antes da correção, os resultados foram insatisfatórios. Isso indica que um *threshold* dinâmico, que se adapte às características dos dados pós-correção, poderá proporcionar uma avaliação mais eficiente e precisa, indicando a necessidade de futuros estudos sobre a escolha de *thresholds*.

Palavras chave

vazamento de dados; reconhecimento de entidades nomeadas; textos legislativos; benchmark; autoaprendizado; português

Abstract

This work addresses data leakage in training Named Entity Recognition (NER) models in Brazilian Portuguese legislative texts, resulting from duplicates and inconsistent annotations, which compromise model evaluation. After correcting this leakage in the UlyssesNER-Br *corpus*, we conducted a new benchmark, comparing the results with previous studies in a more reliable setting. We also re-evaluated a semi-supervised approach using self-learning and active sampling. However, by reusing a fixed threshold, chosen from a cloud of values before the correction, the results were unsatisfactory. This indicates that a dynamic threshold, which adapts to the characteristics of the data post-correction, could provide a more efficient and accurate evaluation, highlighting the need for future studies on threshold selection.

Keywords

data leakage; named entity recognition; legislative texts; benchmark; self-learning; Portuguese

1. Introdução

O conceito de política democrática transcende o simples ato de votar. Ela exige vigilância contínua de uma sociedade civil ativa e de um público bem-informado, ambos responsáveis por controlar e vigiar os atos dos políticos. A transparência, definida como a disponibilização de informações ao público sobre as atividades e decisões das organizações, é um mecanismo essencial para garantir a responsabilização dos agentes públicos, permitindo que os cidadãos avaliem suas ações (Heald, 2006). Iniciativas de governo aberto ampliam essa transparência ao oferecer acesso a documentos e dados relacionados a atividades públicas e oficiais (Lathrop & Ruma, 2010). No entanto, o simples acesso à informação frequentemente não é suficiente para possibilitar uma fiscalização efetiva. Pesquisadores, jornalistas e cidadãos podem se deparar com dificuldades diante da vasta quantidade de projetos de lei, emendas e documentos redigidos em jargão técnico, tornando o processo de análise e compreensão das atividades políticas e legislativas uma tarefa particularmente desafiadora.

No contexto do Processamento de Linguagem Natural (PLN), o Reconhecimento de Entidades Nomeadas (NER, do inglês "*Named Entity Recognition*") desempenha um papel crucial. O NER envolve a identificação de entidades específicas em textos e sua classificação em categorias predefinidas, como pessoas, organizações

DOI: 10.21814/lm.16.2.450
This work is Licensed under a Creative Commons Attribution 4.0 License

e locais. Esse processo pode facilitar a compreensão de documentos ao destacar termos especializados e específicos do domínio, proporcionando uma visão geral mais clara dos textos, como demonstrado nos trabalhos de Sultanum et al. (2018) e Nunes et al. (2019). Além disso, o NER contribui indiretamente para a compreensão de textos ao possibilitar processos de enriquecimento, como a indexação de informações de dicionários, o que facilita a oferta de explicações contextuais e sinônimos. O NER também serve como um estágio inicial em diversas outras tarefas de PLN, como a construção de grafos de conhecimento específicos de domínio e a resolução de correferências (Kalamkar et al., 2022; Cohen, 2005).

Neste estudo investigamos o impacto do vazamento de dados nos resultados de modelos de NER, reavaliando o *benchmark* proposto na literatura e propondo a avaliação de novos modelos para comparação. Além disso, como NER pode ser aprimorado utilizando técnicas semissupervisionadas. Nossa abordagem consiste em uma estratégia de autoaprendizado para ajustar um modelo BERT projetado para NER, utilizando documentos legislativos escritos em português brasileiro como estudo de caso. Para o treinamento e a avaliação, utilizamos o UlyssesNER-Br (Albuquerque et al., 2022), um *corpus* de projetos de lei e consultas legislativas da Câmara dos Deputados do Brasil, explicitamente desenvolvido para NER.

Este artigo é uma versão revisada e ampliada do trabalho anterior (Nunes et al., 2024b). Nesta nova versão, adicionamos uma seção detalhada sobre o vazamento de dados encontrado no *corpus* UlyssesNER-Br, incluindo dados quantitativos e o processo de limpeza realizado. Também introduzimos o novo *corpus*, apresentando alterações tanto no nível de categorias quanto no de tipos. Esta nova versão já se encontra disponível no repositório oficial do UlyssesNER-Br. Além disso, incluímos a avaliação de novos modelos, a comparação entre as versões contaminadas e descontaminadas, o uso de validação cruzada nos experimentos e um novo o *benchmark* refeito agora com testes estatísticos, comparando os resultados com a literatura e analisando o impacto do vazamento de dados nos resultados anteriores.

Dessa forma, as principais contribuições deste artigo são: (i) a avaliação e identificação de diferentes tipos de vazamento de dados no *corpus* UlyssesNER-Br[1]; (ii) a avaliação de modelos de linguagem de compreensão baseados na arquitetura *Transformer*[2], tanto gerais quanto específicos de domínio, em português e multilíngue, resultando no estabelecimento de novos *benchmarks*; e (iii) uma discussão detalhada sobre o NER no contexto legislativo, contrastando os resultados encontrados com os disponíveis na literatura atual.

2. Trabalhos Relacionados

Esta seção explora a literatura sobre NER e o uso de dados não rotulados no processo de treinamento, especificamente em modelos de PLN. A Subseção 2.1 apresenta estudos relevantes sobre NER ao longo do tempo, com ênfase no uso do domínio jurídico em língua portuguesa. Já a Subseção 2.2 aborda o uso de dados não rotulados como uma forma de aumento de dados, concentrando-se especificamente em técnicas de autoaprendizado e aprendizado ativo.

2.1. Reconhecimento de Entidades Nomeadas

Há mais de uma década, Dozier et al. (2010) propuseram um dos sistemas de NER jurídico mais conhecidos, utilizando dados de tribunais dos Estados Unidos, consistindo principalmente em depoimentos, petições e jurisprudência. Os autores utilizaram três métodos para a tarefa de NER, que também podiam ser combinados em sistemas híbridos. Eles também introduziram cinco categorias de entidades, incluindo *jurisdição*, *tribunal*, *título*, *tipo de documento* e *juiz*. De forma semelhante, outros trabalhos exploraram NER no domínio jurídico para outras línguas, como o alemão (Darji et al., 2023; Glaser et al., 2018; Leitner et al., 2019), espanhol (Badji, 2018), grego (Angelidis et al., 2018) e romeno (Păiș et al., 2021). No que se refere ao português, (Santos & Guimarães, 2015) propuseram o primeiro sistema de NER utilizando a arquitetura CharWNN, que emprega uma rede neural perceptron de múltiplas camadas (Santos & Guimarães, 2015). A maioria dos trabalhos sobre NER em textos em português de domínio geral avalia seus modelos utilizando o *corpus* HAREM (Santos et al., 2006), que contém documentos de várias áreas.

No domínio jurídico em língua portuguesa, dois trabalhos recentes introduziram conjuntos de dados em português para NER em textos legais (Luz de Araujo et al., 2018; Albuquerque et al., 2022). Luz de Araujo et al.

[1] https://github.com/ulysses-camara/ulysses-ner-br/tree/main/PL-corpus_v2

[2] Código usado para treinar os modelos e realizar a divisão dos dados está disponível em https://github.com/RafaelOleques/NERLinguamatica2024

(2018) criaram o primeiro conjunto de dados para NER em textos jurídicos brasileiros, chamado LeNER-Br, reunindo 66 documentos legais de tribunais brasileiros, além de treinarem um modelo de rede neural de memória de longa e curta duração com campo aleatório condicional (BiLSTM-CRF) (Lample et al., 2016), resultando em um valor F1 total de cerca de 92% para a classificação de *tokens* e 86% para a classificação de entidades. Albuquerque e colegas (Albuquerque et al., 2022) propuseram, por sua vez, um *corpus* para NER chamado UlyssesNER-Br, composto por projetos de lei e consultas legislativas da Câmara dos Deputados, com 18 tipos de entidades distribuídos em sete categorias. Para validar o *corpus*, os autores implementaram modelos de Campo Aleatório Condicional (CRF, do inglês "*Conditional Random Field*") e Modelo de Markov Oculto (HMM, do inglês "*Hidden Markov Model*") Bird (2006), alcançando uma pontuação F1 de cerca de 80% na análise por categorias e 81% na análise por tipos.

De forma semelhante, três trabalhos recentes exploram contextos jurídicos específicos. Collovini et al. (2019) anotaram manualmente um conjunto de dados policiais usando textos de depoimentos, declarações e interrogatórios, com 916 entidades nomeadas da categoria "Pessoa", alcançando uma pontuação F1 de 89% utilizando BiLSTM-CRF-ELMo. Brito et al. (2023) desenvolveram o CDJUR-BR, um *corpus* do Judiciário brasileiro com entidades específicas do domínio, como *prova*, *pena*, *sentença* e *norma*. Eles obtiveram uma F1-Macro de 0,58 utilizando o modelo BERT (Devlin et al., 2018). Por fim, Correia et al. (2022) desenvolveram um *corpus* com entidades jurídicas de documentos do Supremo Tribunal Federal (STF), anotados por 76 estudantes de Direito. Utilizando BiLSTM-CRF, eles obtiveram um *F1-Weighted* de 93%.

Com base nos avanços em metodologias de ajuste fino e aprimoramentos de modelos, o trabalho proposto por Bonifacio et al. (2020) investigou o impacto do ajuste fino de modelos de linguagem em um grande *corpus* intradomínio de texto não rotulado para NER. Os resultados experimentais revelaram que o ajuste fino dos modelos nestes textos melhorou significativamente o desempenho de NER, particularmente para o modelo BERT, que alcançou resultados de ponta no *corpus* LeNER-Br de textos jurídicos brasileiros. Zanuz & Rigo (2022) introduziram os primeiros modelos BERT ajustados exclusivamente para o português brasileiro, para NER jurídico, alcançando novos resultados de ponta no conjunto de dados LeNER-Br.

2.2. Autoaprendizagem e Aprendizagem Ativa no Treinamento

O aumento de dados tem sido amplamente empregado em diversas tarefas de PLN para aprimorar o desempenho de modelos (Li et al., 2022; Feng et al., 2021; Anaby-Tavor et al., 2020). Um método alternativo para melhorar a qualidade de um *corpus* de treinamento é o uso de dados não rotulados. Em casos onde há uma quantidade substancial de dados não rotulados disponíveis, são utilizadas técnicas semi-supervisionadas (Li et al., 2022; Feng et al., 2021), como autoaprendizagem, aprendizagem ativa e suas variações. Essas técnicas têm demonstrado melhorar os resultados dos modelos em tarefas como classificação (Sha et al., 2022; Alves-Pinto et al., 2021; Mekala & Shang, 2020; Meng et al., 2020; Dong & de Melo, 2019; Dupre et al., 2019) e NER (Gao et al., 2021; Neto & Faleiros, 2021; Helwe & Elbassuoni, 2019; Clark et al., 2018; Tran et al., 2017; Chen et al., 2015).

A abordagem de autoaprendizagem envolve o uso de um *corpus* rotulado para treinar um modelo *professor*, que é utilizado para prever as classes dos dados não rotulados, os quais, por sua vez, são usados para treinar um modelo *aluno* (Dupre et al., 2019). Em algumas estratégias de autoaprendizagem, o modelo aluno pode servir como *professor* para a próxima iteração (Dupre et al., 2019). Embora isso represente uma abordagem tradicional de autoaprendizagem, métodos alternativos, como rótulos fracos, modelos *ensemble* e modificações na função de perda, também podem ser empregados. A aprendizagem ativa segue uma filosofia semelhante, mas incorpora métodos de consulta para selecionar instâncias de interesse para anotação manual. Essas instâncias anotadas são então usadas para treinar o modelo de forma iterativa.

3. *Corpus*

Esta seção apresenta uma visão geral sobre o *corpus* analisado no trabalho, bem como sobre o problema de vazamento de dados descoberto e sua adaptação.

3.1. UlyssesNER-Br

O UlyssesNER-Br (Albuquerque et al., 2022) é um *corpus* em português brasileiro que contém duas fontes de informação e está dividido em dois *corpora*, um para cada fonte de referência. O primeiro *corpus* contém 9.526 sentenças de 150 projetos de lei (*PL-corpus*) da Câmara dos Deputa-

dos do Brasil, enquanto o segundo possui 790 sentenças de solicitações de trabalho (*ST-corpus*).

Ambos os *corpora* do UlyssesNER-Br possuem dois *sub-corpora* com diferentes níveis de entidades: *categoria* e *tipo*. As categorias englobam cinco entidades tradicionais (Albuquerque et al., 2022): "PESSOA", "DATA", "ORGANIZAÇÃO", "EVENTO" e "LOCALIZAÇÃO". Além disso, incluem "FUNDAMENTO" e "PRODUTODELEI" como referências a entidades legislativas. Os tipos, por sua vez, são especializações das categorias, como, por exemplo, "PRODUTOsistema", "PRODUTOprograma" e "PRODUTOoutros" como particularizações da categoria "PRODUTODELEI".

Infelizmente, o *ST-corpus* com solicitações de trabalho não está disponível publicamente, pois se trata de informações internas da Câmara dos Deputados[3], que os autores do UlyssesNER-Br não foram autorizados a compartilhar. Portanto, utilizamos apenas o *PL-corpus* com as informações dos projetos de lei neste estudo.

Em nosso trabalho, utilizamos apenas as entidades de categoria para o processo de autoaprendizagem, uma vez que os autores apontaram que os resultados com categorias e tipos não apresentaram diferenças significativas. Dessa forma, as categorias mostraram-se uma solução mais simples e robusta para o aprendizado do modelo (Albuquerque et al., 2022). A Tabela 1 indica o número de exemplos de cada categoria no *corpus* disponibilizado pelos autores, conforme mostrado na coluna *#Original*. Ressaltamos que a frequência calculada nas tabelas refere-se à ocorrência da entidade completa, e não à frequência de *tokens*.

3.2. Vazamento de dados

Vazamento de dados é um problema crítico em aprendizado de máquina e PLN (Balloccu et al., 2024), no qual os dados do conjunto de teste são inadvertidamente acessados pelo modelo durante o treinamento, resultando em uma superestimação dos resultados e comprometendo a avaliação (Balloccu et al., 2024). Em *corpora* legislativos, o risco de vazamento é particularmente elevado devido à recorrência de termos específicos e padrões linguísticos semelhantes entre os diferentes conjuntos, como é possível ver na Tabela 2.

Identificação e tratamento de instâncias repetidas. Encontramos 70 sentenças com duplicatas e a mesma anotação, e 4 com duplicatas, mas com anotações divergentes. A Tabela 3 apresenta exemplos de ambos os tipos de repetições.

A maioria das instâncias com anotações divergentes foi identificada como casos de anotação parcial, onde as anotações eram essencialmente idênticas, mas com trechos faltantes. No entanto, encontramos um caso de anotação ambígua, em que a mesma entidade nomeada foi marcada como pertencente a classes diferentes em sentenças duplicadas, conforme ilustrado na Tabela 3.

Embora este estudo se concentre apenas no nível de categorias, também analisamos o *sub-corpus* de tipos. Neste *sub-corpus*, identificamos um maior número de sentenças duplicadas, com 512 ocorrências de anotações iguais e 3 com anotações divergentes, conforme apresentado na Tabela 4. Nesse caso, as anotações divergentes eram essencialmente iguais, mas apresentavam partes faltantes.

Além disso, identificamos sobreposições entre os conjuntos originalmente disponibilizados para treino, validação e teste, como mostrado na Tabela 5. Essa tabela revela duplicatas dentro de cada conjunto, o que pode levar ao reforço de padrões específicos. Observamos também sobreposições entre diferentes conjuntos, o que pode resultar em uma superestimação dos resultados, uma vez que o modelo pode ser treinado com dados que estavam presentes em mais de um conjunto.

Após a identificação de instâncias repetidas, consultamos um especialista para revisar as anotações, garantindo maior precisão ao conjunto de dados. Nas instâncias em que a anotação original foi considerada inadequada, seja por anotação parcial ou ambígua, o especialista atribuiu uma nova anotação mais adequada ao contexto. Já para os casos em que a anotação era essencialmente igual entre as ocorrências repetidas, mantivemos apenas a primeira ocorrência no *corpus*. Na Subseção 3.3, apresentamos as estatísticas do novo conjunto gerado e uma descrição atualizada do *corpus*.

3.3. *Corpus* atualizado

Após a implementação das estratégias de mitigação de vazamento de dados e a padronização das anotações, realizamos uma análise estatística detalhada do *corpus* atualizado. Nesta subseção, discutimos a nova divisão dos dados, que foi feita tanto para a abordagem de *holdout* quanto para a validação cruzada.

[3] https://github.com/Convenio-Camara-dos-Deputados/ulyssesner-br-propor/tree/main/Corpora

	Entidades			**Sentenças**		
Classe	**#Original**	**#Filtrado**	**Δ**	**#Original**	**#Filtrado**	**Δ**
DATA	603	427	-176	522	346	-176
EVENTO	23	23	0	21	21	0
FUNDAMENTO	721	716	-5	522	519	-3
LOCAL	615	607	-8	325	319	-6
ORGANIZACAO	610	598	-12	469	460	-9
PESSOA	861	847	-14	545	536	-9
PRODUTODELEI	330	319	-11	277	267	-10
Soma	3.763	3.537	-226	2.681	2.468	-213

Tabela 1: Número de instâncias e número de sentenças que possuem cada classe no *corpus* UlyssesNER-Br no nível de categoria, antes e após a filtragem proposta neste trabalho.

Padrão	**Categorias**	**Tipos**
sala das sessões , em de de [ANO]	88	79
projeto de lei nº , de [ANO] (do sr .	42	44

Tabela 2: Exemplos de padrões identificados no *PL-corpus*. A tabela apresenta as frases padrão, seguidas pela quantidade de repetições nos *sub-corpora* de categorias e tipos.

Tipo	**Elemento**	**Exemplos**
Duplicada	Sentença:	câmara dos deputados projeto de lei nº , de 2019 (do sr .
	Anotação:	B-ORGANIZACAO I-ORGANIZACAO I-ORGANIZACAO O O O O O O B-DATA O O O O
Parcial	Sentença:	sala das sessões , em de agosto de 2019 .
	Anotação 1:	O O O O O O B-DATA I-DATA I-DATA O
	Anotação 2:	O O O O O O O O O O
Ambígua	Sentença:	recentemente , foi publicada a lei nº 13.819 , de 2019 6 , que instituiu a política nacional de prevenção da automutilação e do suicídio , e trouxe diversas inovações ao ordenamento jurídico , no contexto da prevenção desse agravo
	Anotação 1:	O O O O O B-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO O O O O O B-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO O O O O O O O O O O O O O O O O O O O
	Anotação 2:	O O O O O B-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO I-FUNDAMENTO O O O O O B-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI I-PRODUTODELEI O O O O O O O O O O O O O O O O O O O

Tabela 3: Exemplos de anotações duplicadas, parciais e ambíguas encontradas no *sub-corpus* de categorias.

Tipo	Elemento	Exemplos
Duplicada	Sentença:	sala das sessões , em de de 2019 .
	Anotação:	O O O O O O O B-DATA O
Parcial	Sentença:	câmara dos deputados projeto de lei nº , de 2011 (do sr .
	Anotação 1:	O O O O O O O O O B-DATA O O O O
	Anotação 2:	B-ORGgovernamental I-ORGgovernamental I-ORGgovernamental O O O O O O B-DATA O O O O

Tabela 4: Exemplos de anotações duplicadas e parciais encontradas no *sub-corpus* de tipos.

Conjuntos	#Instâncias
Treino e Validação	
#Sentenças[1]	13
#Treino[2]	95
#Validação[2]	30
Treino e Teste	
#Sentenças[1]	22
#Treino[2]	128
#Teste[2]	33
Validação e Teste	
#Sentenças[1]	5
#Validação[2]	21
#Teste[2]	10
Treino, Validação e Teste	
#Sentenças[1]	5
#Treino[2]	85
#Validação[2]	21
#Teste[2]	10

Tabela 5: Número de instâncias duplicadas com entidades nomeadas entre os conjuntos originais do UlyssesNER-Br. Ambos os *sub-corpus* possuem os mesmos valores. [1]Cada sentença é contada somente uma vez, independente do número de cópias. [2]Cada sentença é contadas N vezes, sendo N o número de cópias que ela possui.

Distribuição das entidades no *corpus*. As Tabelas 1 e 6 mostram a distribuição das entidades em ambos os *sub-corpora*. Na coluna *#Original* estão o número de instâncias originais de cada classe; na coluna *#Filtrado*, o número de instâncias após a filtragem para remoção de sentenças duplicadas; e na coluna Δ, a diferença entre os dois conjuntos. É relevante observar que, embora o *sub-corpus* de tipos contenha entidades especializadas em comparação com o *sub-corpus* de categorias, não foi realizada a anotação aninhada entre os níveis, o que resultou em discrepâncias nas estatísticas entre eles.

Observamos uma redução no número de sentenças, com uma diminuição de 319 instâncias no nível de categorias e de 787 instâncias no nível de tipos. As Tabelas 1 e 6 fornecem detalhes sobre o impacto dessa redução nas sentenças de cada classe.

A análise das tabelas revela que a filtragem resultou em uma redução na quantidade de *tokens* e sentenças no *corpus*. Embora a distribuição das entidades não tenha mostrado variações significativas na frequência das classes —com as classes majoritárias continuando a dominar e as minoritárias permanecendo menos frequentes—, a filtragem reduziu o número total de sentenças disponíveis para análise. No entanto, a remoção de duplicatas pode ter contribuído para uma melhoria na qualidade geral dos dados. A comparação entre os *sub-corpus* de categorias e tipos revela diferenças na distribuição que podem impactar o treinamento e a avaliação dos modelos. Assim, é fundamental considerar essas diferenças e seus potenciais efeitos na performance dos modelos, além de avaliar a necessidade de técnicas adicionais para o balanceamento de dados.

4. Metodologia

Nesta seção, detalhamos a metodologia utilizada neste trabalho. Descrevemos o *corpus* não rotulado empregado no processo de autoaprendizagem, a divisão do *corpus* para o treinamento dos modelos, os modelos utilizados, os critérios de avaliação e o pipeline de autoaprendizagem implementado.

4.1. Ementas da Câmara dos Deputados do Brasil

As ementas dos projetos de lei são obtidas por meio da API da Câmara dos Deputados[4] abrangendo de 1991 a 2022. Essas ementas são, então, segmentadas em frases usando a expressão regular "`. (?=[A-Za-z])`" para identificar pontos

[4] https://dadosabertos.camara.leg.br/swagger/api.html

	Entidades			Sentenças		
Classe	**#Original**	**#Filtrado**	**Δ**	**#Original**	**#Filtrado**	**Δ**
DATA						
DATA	629	383	-246	541	309	-232
EVENTO						
EVENTO	27	18	-9	25	17	-8
FUNDAMENTO						
FUNDapelido	191	157	-34	176	145	-31
FUNDlei	529	442	-87	463	386	-77
FUNDprojetodelei	18	13	-5	14	11	-3
LOCAL						
LOCALconcreto	509	418	-91	285	232	-53
LOCALvirtual	62	45	-17	47	36	-11
ORGANIZACAO						
ORGgovernamental	460	387	-73	375	312	-63
ORGnaogovernamental	132	104	-28	105	83	-22
ORGpartido	31	27	-4	23	21	-2
PESSOA						
PESSOAcargo	304	260	-44	259	220	-39
PESSOAgrupocargo	161	140	-21	119	106	-13
PESSOAindividual	401	337	-64	368	310	-58
PRODUTODELEI						
PRODUTOprograma	64	53	-11	56	47	-9
PRODUTOsistema	17	15	-2	16	14	-2
PRODUTOoutros	257	207	-50	217	175	-42
Soma	3.792	3.006	-786	3.089	2.424	-665

Tabela 6: Classes e instâncias no *corpus* UlyssesNER-Br no nível de tipos, antes e após a filtragem proposta neste trabalho.

seguidos por letras, dividindo o texto em frases. Escolhemos essa expressão regular porque o domínio do texto legislativo inclui construções como "`Art. 123`", onde o ponto faz parte do nome do artigo e não indica o fim de uma frase.

Esse processo produziu 428.573 frases, com uma contagem média de palavras de 32,52 e um desvio padrão de 42,64. Ao obter frases, excluímos as que já estavam presentes no *corpus* UlyssesNER-Br, certificando-nos de incluir apenas frases distintas para evitar a inserção de contaminação ou *overfitting*. Todas essas frases não tinham informações NER.

4.2. Divisão do *corpus*

Para treinar e validar nossa abordagem, utilizamos um método de divisão de validação cruzada com cinco partições, inspirado no artigo original do UlyssesNER-Br (Albuquerque et al., 2022). A principal distinção é que usamos divisão estratificada, uma modificação influenciada pela abordagem de Sechidis et al. (2011). Também introduzimos uma etapa adicional de pré-processamento que gera uma lista equivalente ao número de entidades distintas possíveis. Dentro dessa lista, cada posição é marcada com um valor 1, se a entidade correspondente estiver presente na sentença, e com 0, caso contrário. Essa modificação nos permitiu estratificar a divisão com base na presença de cada entidade.

É importante destacar a relevância desse passo de estratificação, especialmente devido ao desequilíbrio substancial de classes no *corpus*. Esse desequilíbrio é evidente ao examinar exemplos de classes minoritárias e majoritárias, como "Eventos", com apenas 21 instâncias, em contraste com "Pessoa", que possui 847 instâncias no *sub-corpus* de categorias.

4.3. Modelos

Descrevemos brevemente os modelos *Transformers* mais proeminentes para a língua portuguesa. Utilizamos os modelo com as arquiteturas BERT, DeBERTa e RoBERTa para a tarefa de NER e o modelo SBERT para a amostragem ativa.

BERTimbau é um modelo BERT pré-treinado e ajustado para o português brasileiro (Souza et al., 2020). Até onde sabemos, o BERTimbau é o estado da arte em Reconhecimento de Entidades Nomeadas, similaridade textual de sentenças e reconhecimento de inferências textuais em português brasileiro. Este modelo possui duas versões principais, *large* e *base*. No texto, quando não referenciado explicitamente, o modelo referido é o BERTimbau-*large*.

XLM-RoBERTa (XLM-R) (Conneau et al., 2020) é um modelo multilíngue que tem a capacidade de processar 100 idiomas diferentes sem a necessidade de incorporações específicas para cada língua. Ele combina abordagens do RoBERTa (Liu et al., 2019) dentro do framework do XLM (Conneau et al., 2020), permitindo que o modelo identifique o idioma a partir dos *tokens* de entrada. O foco do XLM-R está na modelagem de linguagem mascarada para sentenças monolíngues, excluindo a técnica de modelagem de linguagem de tradução utilizada no XLM.

Albertina PT-BR (Rodrigues et al., 2023) é uma versão do modelo de linguagem Albertina projetada para o português brasileiro. Baseado na arquitetura DeBERTa (He et al., 2021), este modelo possui uma licença permissiva e utiliza um *corpus* multilíngue com 36 bilhões de *tokens*, ampliando sua capacidade de entendimento e geração de texto em português.

BERTikal é um modelo BERT ajustado para a linguagem jurídica brasileira (Polo et al., 2021). A base utilizada foi o BERTimbau base cased (Souza et al., 2020), empregando um objetivo de Modelagem de Linguagem Mascarada (MLM). Os documentos utilizados para o treinamento do BERTikal incluíram publicações e petições de tribunais brasileiros, além de documentos legais mais extensos, principalmente oriundos do Tribunal de Justiça de São Paulo (TJSP).

Jurisbert e **BERTimbaulaw** são desenvolvidos para o português brasileiro, como proposto por Viegas et al. (2023). Os modelos utilizaram informações resumidas coletadas de tribunais brasileiros por meio de web scraping. Enquanto o Jurisbert foi treinado do zero, após o pré-treinamento do BERT base (Devlin et al., 2018), o BERTimbaulaw aproveitou o pré-treinamento do BERTimbau base (Souza et al., 2020).

LegalBERT-pt (Silva et al., 2021) é um modelo BERT pré-treinado com *corpora* jurídicos brasileiros. Os autores realizaram o pré-treinamento de dois tipos de modelos: LegalBert-pt SC e LegalBert-pt FP. O modelo SC foi elaborado a partir do treinamento de um modelo BERT do zero, com a mesma configuração do BERTimbau base. Por outro lado, o modelo FP pré-treinou um BERTimbau-Base utilizando *corpora* específicos do domínio jurídico. Os *corpora* utilizados no estudo englobam diversos casos judiciais brasileiros e, para o modelo SC, artigos da Wikipédia em português, totalizando 1.500.000 documentos legais e um vocabulário de 36.345 palavras.

RoBERTaLexPT (Garcia et al., 2024) é um modelo baseado no RoBERTa (Liu et al., 2019), que foi aprimorado com o *corpus* LegalPT (Garcia et al., 2024), especializando-o para a linguagem jurídica portuguesa. O *corpus* utilizado na fase de pré-treinamento abrange uma variedade de conjuntos de dados, predominantemente dados jurídicos, mas também incluindo informações gerais, em português brasileiro e português europeu.

Legal-XLM-RoBERTa e **Legal-portuguese-RoBERTa** são modelos adaptados a partir do pré-treinamento do XLM-R (Conneau et al., 2020) para o domínio jurídico (Niklaus et al., 2024). Ambos os modelos foram treinados utilizando o *corpus* Multilegalpile (Niklaus et al., 2024), sendo que o primeiro foi treinado com todo o *corpus*, enquanto o segundo focou especificamente na porção em português.

SBERT (Reimers & Gurevych, 2019) é uma modificação dos modelos BERT que utiliza redes siamesas e *triplets* para obter *embeddings* contextuais referentes a sentenças inteiras. Para gerar *embeddings* para textos em português, utilizamos a versão multilíngue do SBERT (Reimers & Gurevych, 2020), que está disponível no Hugging Face Hub[5] na técnica de amostragem ativa.

4.4. Avaliação de modelos

Com a atualização do *corpus*, realizamos uma nova avaliação dos modelos contextuais em português, multilíngues e específicos do domínio legal, previamente feita por Garcia et al. (2024). Embora o estudo original apresente vários hiperparâmetros utilizados, os valores finais de alguns que passaram por otimização, como *batch size* e *learning rate*, não foram informados. Por isso, optamos por utilizar os mesmos hiperparâmetros

[5] https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1

do nosso trabalho anterior (Nunes et al., 2024b), que, em grande parte, estão alinhados com os de Garcia et al. (2024).

Entretanto, nossa avaliação difere da realizada anteriormente, pois, para evitar que resultados otimistas ou pessimistas sejam gerados aleatoriamente devido à abordagem *holdout*, optamos pela validação cruzada estratificada com 5 partições. Esse tipo de análise permite obter uma avaliação mais justa do modelo por meio da média das partições, além de proporcionar uma noção maior de robustez com base na análise do desvio padrão. Por fim, utilizamos as médias das partições em cada métrica para testes estatísticos (Gomes et al., 2021; Nunes et al., 2024a), a fim de determinar se as diferenças encontradas entre os modelos são significativas.

4.5. Autoaprendizado

A Figura 1 ilustra o processo de autoaprendizagem. Seguimos esse *pipeline* em cada iteração da validação cruzada e divisão de amostras. O *pipeline* começa com o treinamento do primeiro classificador usando os dados de treinamento (1 a 4). Uma vez que nosso *corpus* de ementas contém uma grande quantidade de dados, a técnica de amostragem é usada para otimizar o tempo de treinamento ao longo das iterações de autoaprendizagem (5 a 6).

Inspirados pela amostragem dinâmica de Sha et al. (2022), nossa técnica de amostragem começa com uma amostragem aleatória do *corpus* de ementas, produzindo $N\%$ dos dados não rotulados com um mínimo de 2.000 amostras. Em seguida, usamos a amostragem baseada em diversidade (Tran et al., 2017; Chen et al., 2015) aplicando similaridade de cosseno aos dados amostrados em relação aos dados de treinamento. Posteriormente, obtemos as amostras mais dissimilares em um total de $K\%$ dos dados amostrados, com um mínimo de 1.000, que são usados no modelo para prever os *labels* de NER para cada sentença. Os *embeddings* usados para calcular a similaridade de cosseno são gerados pelo SBERT, devido à sua capacidade de reconhecer características importantes de sentenças.

Após a etapa de amostragem ativa, aplicamos o classificador NER a cada sentença (7 a 9.a). Para determinar quais sentenças devem ser adicionadas aos dados de treinamento, medimos a confiança média da predição das entidades previstas (Gao et al., 2021) (10). Se a confiança média for igual ou superior a um limite, a sentença é utilizada nos dados de treinamento (11.a a 12.a) e removida do *corpus* de ementas (11.b a 12.b); caso contrário, ela é mantida no *corpus* de ementas e não é usada para o treinamento. Posteriormente, o *pipeline* é reiniciado usando o novo conjunto de treinamento para treinar um novo modelo e repetir todo o processo.

O processo é interrompido quando uma condição de parada antecipada é encontrada com base no F1 geral. Descrevemos os hiperparâmetros na Seção 5.3. Também implementamos um critério de parada antecipada no qual nenhum dado foi adicionado ao treinamento ou se o conjunto não rotulado ficou vazio (ou seja, todos os dados disponíveis foram utilizados). Ocasionalmente, devido à abordagem de amostragem aleatória, é possível que os dados selecionados aleatoriamente não contenham exemplos adequados para o treinamento, resultando em nenhuma adição ao conjunto de treinamento. Nesses casos, implementamos um critério de espera que permite um máximo de W novas amostragens antes de encerrar o processo de autoaprendizagem. Cada uma dessas novas amostragens usa sementes aleatórias diferentes para gerar conjuntos distintos, visando resolver possíveis problemas com a seleção aleatória inicial.

5. Avaliação experimental

Nesta seção, apresentamos uma avaliação experimental da abordagem proposta. Descrevemos o *setup* do experimento, incluindo o hardware utilizado e o ambiente de desenvolvimento. Também detalhamos os hiperparâmetros do nosso modelo, o treinamento com autoaprendizagem e as métricas usadas para a avaliação.

5.1. Setup

Utilizamos um computador com uma GPU Nvidia GeForce RTX 4090 e 64 GB de RAM para o treinamento e avaliação dos modelos, bem como para a obtenção dos dados da API da BCoD[6]. Optamos pela linguagem de programação Python 3.7.6 devido à sua vasta gama de bibliotecas para aprendizado de máquina e processamento de linguagem natural.

5.2. Hiperparâmetros do modelo

Calibramos o modelo com base em estudos anteriores (Zanuz & Rigo, 2022; Bonifacio et al., 2020). Utilizamos os modelos descritos na Seção 4.3, treinados com o *PL-corpus* para a tarefa de Reconhecimento de Entidades Nomeadas. Para

[6] https://dadosabertos.camara.leg.br/swagger/api.html

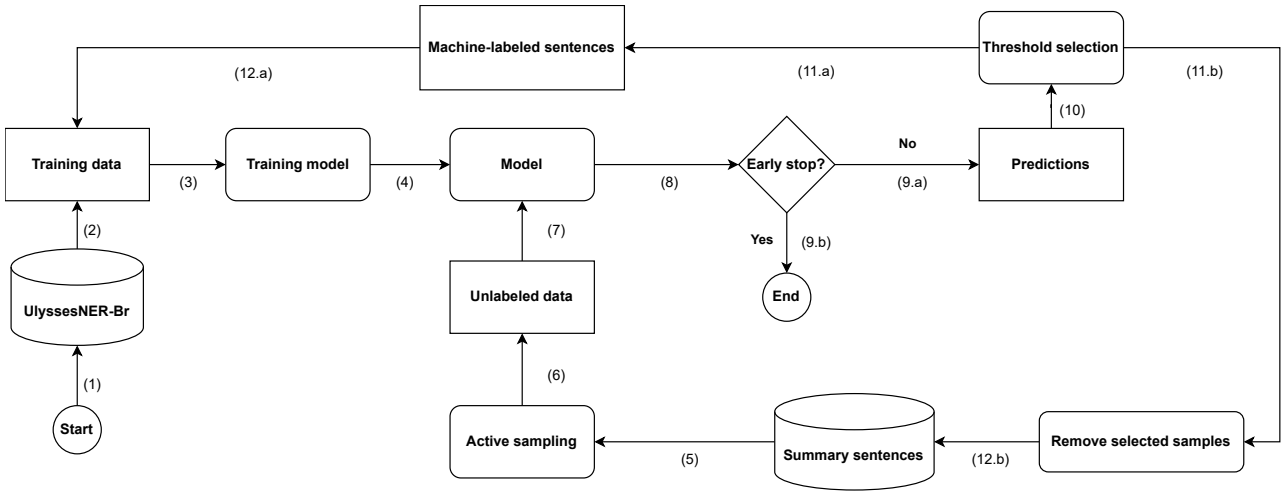


Figura 1: *Pipeline* da técnica de autoaprendizagem (Nunes et al., 2024b).

o treinamento dos modelos, utilizamos a API de treino da Huggingface[7], configurando o comprimento máximo de 512 *tokens* por sentença, com a aplicação de *padding* e truncamento conforme necessário.

Os hiperparâmetros utilizados para construir o modelo foram: *evaluation_strategy = epochs*, *save_total_limit = 5*, *learning_rate = 2e-05*, *weight_decay = 0.01*, *otimizador = Adam com betas = (0.9, 0.999) e epsilon = 1e-08*. Os demais parâmetros não especificados seguiram os valores padrão do modelo. Também configuramos o *save_strategy* para *epoch* e usamos o F1 micro como a métrica escolhida para o melhor modelo.

5.3. Hiperparâmetros do autoaprendizado

Para a amostragem ativa, utilizamos $N = 0.05$ para a porcentagem de amostras aleatórias, $K = 0.6$ para a porcentagem de amostras dissimilares e 42 como a primeira *seed* para a amostragem aleatória. Definimos $W = 5$ como o número máximo de novas amostragens aleatórias para aumentar a quantidade de dados de treinamento. Também aplicamos uma paciência igual a 3 para aguardar um aumento no F1 geral durante as iterações de autoaprendizado. Usamos 0,99 como *threshold* de confiança médio para as previsões seguindo o valor encontrado por Nunes et al. (2024b).

5.4. Métricas

Utilizamos a biblioteca seqeval[8] para calcular as métricas. Um aspecto interessante dessa biblioteca é que ela computa os resultados com base na sequência de tags para cada entidade (começando com um "B-TAG" seguido de "I-TAG"); para uma sentença completa, é essencial reconhecer claramente uma entidade, em vez de apenas um *token* específico.

Calculamos as seguintes métricas: F1-Score, precisão, revocação e acurácia (somente para o caso geral). Optamos pelo F1-Score como a principal métrica para nossas análises, pois desejamos equilibrar a previsão correta da classe positiva e a precisão dessa classe.

5.5. Métodos de avaliação de desempenho

A escolha da técnica de validação é fundamental para garantir uma avaliação precisa do desempenho do modelo. Para comparar as diferentes versões dos *sub-corpora*, utilizamos a Validação Cruzada Repetida Estratificada com K-partições, que proporciona uma análise mais robusta e reduz a variação nos resultados. Já a Validação Cruzada Estratificada com K-partições e Retenção foi aplicada no processo de autoaprendizagem, permitindo ajustar o modelo de forma eficaz e selecionar o *threshold* ideal com base no desempenho das diferentes partições.

Validação Cruzada Repetida Estratificada com K-partições (*Repeated Stratified K-fold Cross Validation*). Para avaliar o impacto da descontaminação em *sub-corpus*, essa técnica é fundamental para garantir resultados mais confiáveis. A validação cruzada estratificada assegura a distribuição equilibrada das classes em cada partição, enquanto a repetição do processo reduz a dependência de divisões aleatórias, resultando em uma média e desvio padrão mais representativos do desempenho do modelo (Kim, 2009; Machado et al., 2015).

[7] https://huggingface.co/docs/transformers/en/main_classes/trainer

[8] https://github.com/chakki-works/seqeval

Essa abordagem proporciona uma avaliação mais robusta, permitindo comparações diretas mais precisas entre diferentes versões do *corpus* (Kim, 2009). A repetição da validação também facilita a realização de testes estatísticos mais sólidos, favorecendo uma análise objetiva dos efeitos da descontaminação, com menor influência de variações aleatórias.

Validação Cruzada Estratificada com K–partições e Retenção (*Holdout Stratified K-Fold Cross-Validation*). Seguindo a abordagem previamente descrita, realizamos um *benchmark* ajustando o modelo BERT para a tarefa de NER utilizando um *corpus* legislativo em português (Albuquerque et al., 2022). O treinamento começou com o ajuste fino inicial do modelo através de uma validação cruzada estratificada de cinco partições, a fim de determinar o valor ideal do *threshold* de autoaprendizagem, conforme descrito na Subseção 5.3. Em seguida, aplicamos o *threshold* em cada partição, conforme mencionado na Seção 4.5.

É importante destacar que o autoaprendizado gera métricas para os classificadores em cada iteração. Em vez de depender apenas do classificador final do *pipeline*, adotamos uma abordagem mais robusta, selecionando as métricas do classificador que apresentou o melhor F1-Score geral. Dentro de cada partição, o *threshold* para o desempenho ideal foi determinado com base no F1-Score mais alto. Para estabelecer o melhor F1-Score para o modelo final, selecionamos o melhor F1-Score em torno das 5 partições, usando a média e o desvio padrão para identificar o *threshold* que consistentemente produziu resultados superiores. Utilizamos esse *threshold* durante a fase de treinamento *holdout*.

5.6. Testes estatísticos

Para ter uma melhor comparação dos modelos, aplicamos testes estatísticos. Para isso, foi utilizada a média do F1-Score de cada classe como um vetor para representar a distribuição do resultado do modelo, seguindo a metodologia utilizada por Gomes et al. (2021) e Nunes et al. (2024a). Com isso, aplicamos o teste de Shapiro-Wilk (Shapiro & Wilk, 1965) para verificar a normalidade, e identificamos que as distribuições não são normais em ambos os *sub-corpora*, tanto com contaminação quanto sem.

Diante da não normalidade dos dados, utilizamos o teste de Friedman (Friedman, 1937) em combinação com o teste *post-hoc* de Nemenyi (Nemenyi, 1963) para comparar os modelos dentro da mesma versão de um *sub-corpus* específico, avaliando simultaneamente múltiplos modelos. O teste de Wilcoxon Signed-Rank (Wilcoxon, 1945), por sua vez, foi aplicado para comparar o desempenho de cada modelo entre as versões contaminada e descontaminada de um mesmo *sub-corpus*, realizando uma comparação direta entre as versões de um único modelo.

6. Resultados e discussão

Nesta seção, analisamos o desempenho dos modelos empregados em nossos experimentos com o *corpus* atualizado, seguindo o nível de detalhamento utilizado em trabalhos anteriores (Albuquerque et al., 2022, 2023; Garcia et al., 2024; Nunes et al., 2024b), focando nas métricas macro. Incluímos testes estatísticos para identificar possíveis diferenças significativas entre os resultados e comparamos nossos achados com a literatura existente. Além disso, consideramos o impacto potencial do vazamento de dados, que pode influenciar a interpretação dos resultados obtidos. Por fim, apresentamos os resultados da aplicação de autoaprendizagem no *sub-corpus* de categorias. Devido à importância e ao custo computacional da validação utilizada para avaliar a contaminação e descontaminação dos *sub-corpora*, limitamos a avaliação à versão descontaminada (Nunes et al., 2024b), pois não encontramos diferença significativa entre as versões.

6.1. Análise dos resultados do *sub-corpus* de categorias

Analisando a Tabela 7 com os experimentos sobre o *corpus* descontaminado, identificamos uma variação significativa no F1-Score entre o melhor e o pior modelo, com médias que vão de 69,59% até 83,42%. Essa variação indica que, apesar de alguns modelos terem desempenho destacado, a eficácia em NER pode ser altamente variável.

Ao examinarmos os modelos com melhor desempenho, não encontramos um padrão claro que indique que modelos com pré-treinamento continuado no contexto legal superem aqueles com pré-treinamento específico para o português ou multilíngues. Os dois modelos que compartilham a arquitetura XLM-RoBERTa (Conneau et al., 2019) apresentam médias 81,99% (XLM-R-base, desvio padrão 2,41) e 82,51% (RoBERTaLexPT, desvio padrão 1,83), englobando tanto modelos em português com dados legais quanto modelos multilíngues. Em relação a esta arquitetura, vemos uma tendência de o resultado melhorar com dados mais próximos aos da tarefa, tanto com uma maior média quanto com um menor desvio padrão, seja pelo idioma ou seja pelo domínio.

Modelo	Precisão	Revocação	F1
BERTimbau-*large*	80,71 ± 3,22	86,38 ± 2,45	83,42 ± 2,50
RoBERTaLexPT	79,33 ± 2,51	86,00 ± 1,93	82,51 ± 1,83
BERTimbau-*base*	79,70 ± 2,31	85,50 ± 1,95	82,49 ± 1,92
LegalBert-pt_FP	79,34 ± 3,37	85,60 ± 2,13	82,33 ± 2,58
XLM-R-*base*	78,59 ± 3,40	85,76 ± 1,85	81,99 ± 2,41
BERTimbaulaw	78,20 ± 2,47	85,39 ± 2,33	81,62 ± 2,10
Jurisbert	75,21 ± 2,90	82,10 ± 2,47	78,48 ± 2,34
LegalBert-pt_SC	69,45 ± 3,38	76,95 ± 3,25	72,98 ± 3,00
BERTikal	66,34 ± 3,36	73,25 ± 2,87	69,59 ± 2,78

Tabela 7: Resultados da avaliação dos modelos no ***sub-corpus*** **descontaminado de categorias**.

Modelo	Precisão	Revocação	F1
BERTimbau-*large*	81,57 ± 2,31	87,06 ± 2,25	84,21 ± 2,06
BERTimbau-*base*	81,01 ± 2,29	86,10 ± 2,12	83,46 ± 1,89
LegalBert-pt_FP	80,31 ± 2,42	85,92 ± 2,42	83,00 ± 2,07
XLM-R-*base*	79,32 ± 2,14	85,56 ± 2,52	82,31 ± 2,10
RoBERTaLexPT	78,84 ± 3,11	85,93 ± 2,44	82,21 ± 2,47
BERTimbaulaw	78,93 ± 2,34	85,30 ± 2,62	81,97 ± 2,17
Jurisbert	76,56 ± 2,82	82,78 ± 2,33	79,52 ± 2,16
LegalBert-pt_SC	70,69 ± 2,79	77,57 ± 2,71	73,93 ± 2,15
BERTikal	69,15 ± 2,74	74,77 ± 2,67	71,83 ± 2,46

Tabela 8: Resultados da avaliação dos modelos no ***sub-corpus*** **contaminado de categorias**.

Em contraste, dois modelos apresentam médias em torno de 70%: BERTikal (média 69,59%) e LegalBert-pt SC (média 72,98%) com os maiores desvios padrões (2,78 e 3,00, respectivamente). Neste grupo observamos somente modelos BERT com pré-treino em textos legais, demonstrando que nem sempre o modelo com o pré-treino dentro do domínio dos textos da tarefa possui o melhor desempenho.

O modelo BERTimbau, que já havia demonstrado resultados de estado da arte para NER em português e no domínio legal (Souza et al., 2020; Zanuz & Rigo, 2022; Nunes et al., 2024b; Garcia et al., 2024), exibe uma média de desempenho alta em suas versões *base* e *large* com, respectivamente, 82,49% e 83,42%. É interessante destacar que o modelo *base* do BERTimbau alcançou resultados comparáveis à versão *large* com uma diferença de F1-Score de 0,93, além de um menor desvio padrão, apresentando uma tendência de maior robustez para o aprendizado desta tarefa. O que nos leva a sugerir que modelos menores ainda são competitivos para a tarefa de NER em detrimento de suas versões maiores.

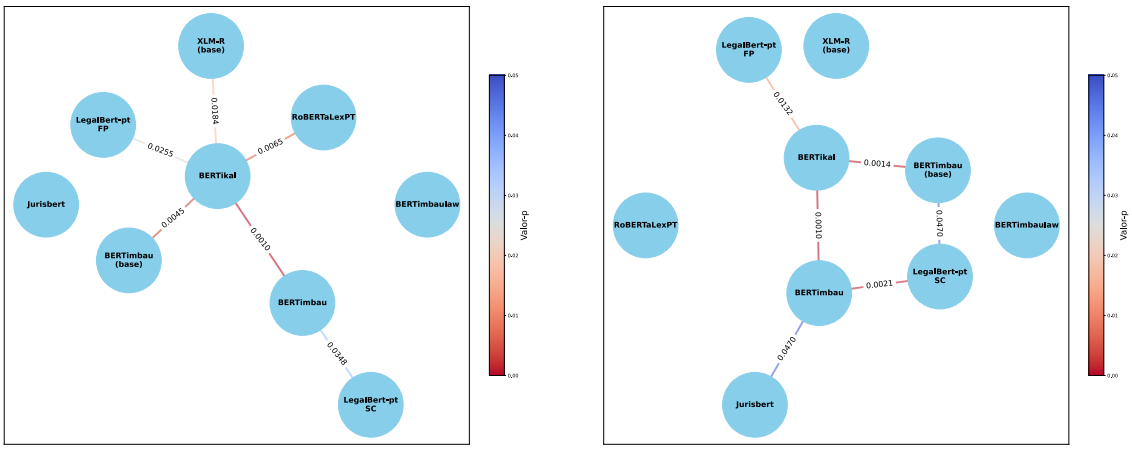
Ainda que o modelo BERTimbau *large* tenha obtido a maior média, observamos que ao levar em conta o *trade-off* entre média e desvio padrão, o modelo RoBERTaLexPT se sobressai, apresentando uma média próxima com diferença de 0,91 e menor desvio padrão com diferença de 0,67. Essa maior estabilidade no desvio padrão sugere que o modelo RoBERTaLexPT pode tender a ser mais adequado ao contexto legal, onde a consistência nos resultados é fundamental.

Em relação à comparação do desempenho dos modelos aplicados à versão descontaminada do *sub-corpus* de categorias, a Figura 2a mostra que, com o teste de Friedman seguido do *post-hoc* de Nemenyi, o modelo com maior diferença estatística em comparação aos demais é o BERTikal, que apresenta a menor média de F1-Score entre os modelos analisados (ver Tabela 7). Além disso, o BERTimbau, que obteve a maior média, demonstrou uma diferença significativa não apenas em relação ao BERTikal, mas também ao LegalBert-pt_sc, que ficou com o segundo pior F1-Score. Esses resultados indicam uma tendência favorável ao BERTimbau como o modelo preferido.

No entanto, é importante destacar que, embora o BERTimbau tenha mostrado um desempenho superior nos testes, sendo significativamente melhor que dois modelos (BERTikal e LegalBert-pt_SC), o valor-p em relação ao LegalBert-pt_SC é ligeiramente mais elevado, o que sugere uma diferença com menor confiança, apesar de ainda ser significativa. Mesmo assim, o BERTimbau e o RoBERTaLexPT apresen-

Modelo	F1 (Contaminado)	F1 (Descontaminado)	Δ F1	Valor-p
BERTimbau-*large*	84,21 ± 2,06	83,42 ± 2,50	-0,79	0,578125
RoBERTaLexPT	82,21 ± 2,47	82,51 ± 1,83	+0,30	0,687500
BERTimbau-*base*	83,46 ± 1,89	82,49 ± 1,92	-0,97	0,578125
LegalBert-pt_FP	83,00 ± 2,07	82,33 ± 2,58	-0,67	0,937500
XLM-R-*base*	82,31 ± 2,10	81,99 ± 2,41	-0,32	0,296875
BERTimbaulaw	81,97 ± 2,17	81,62 ± 2,10	-0,65	0,578125
JurisBert	79,52 ± 2,16	78,48 ± 2,34	-1,04	0,578125
LegalBert-pt_SC	73,93 ± 2,15	72,98 ± 3,00	-0,95	0,687500
BERTikal	71,83 ± 2,46	69,59 ± 2,78	-2,24	0,115851

Tabela 9: Resultados da avaliação dos modelos no ***sub-corpus* de categorias** com e sem contaminação, incluindo a diferença na métrica F1 e o valor-p obtido pelo teste de Wilcoxon Signed-Rank.



(a) Versão descontaminada. **(b)** Versão contaminada.

Figura 2: Gráficos de rede que ilustram a comparação entre os modelos, utilizando o ***sub-corpus* de categorias**. As arestas conectam os modelos que apresentam diferenças significativas, sendo o valor-p mapeado para cor da aresta. Nodos desconectados não tem diferenças significativas para os demais. As figuras com os resultados completos estão disponíveis no Apêndice A.

tam desempenhos bastante semelhantes, com diferenças estatísticas próximas. Como discutido, o RoBERTaLexPT se destaca pela maior estabilidade, apresentando uma média de F1-Score similar à do BERTimbau, mas com um desvio padrão menor, o que sugere maior consistência nos resultados. Esses fatores indicam que a escolha entre os dois modelos pode depender de aspectos como variabilidade dos resultados, estabilidade e o tamanho do modelo. O RoBERTaLexPT pode ser mais adequado em cenários que exigem um modelo mais leve e estável. Por outro lado, o BERTimbau, devido ao seu treinamento com uma maior diversidade de dados e ao seu uso mais amplo, oferece uma capacidade de generalização superior, tornando-o uma escolha robusta, especialmente em tarefas com dados variados ou quando há uma quantidade limitada de dados disponíveis.

Por fim, as análises sugerem que a performance dos modelos de NER não depende unicamente do tamanho ou do tipo de pré-treinamento. A relação entre o alinhamento com o domínio e a arquitetura é crucial, e modelos menores podem, em alguns casos, apresentar menor variância e resultados estáveis. Essa discussão ressalta a necessidade de uma avaliação cuidadosa ao selecionar modelos para tarefas de NER, onde a interação entre arquitetura, pré-treinamento e alinhamento ao domínio são fatores importantes.

Comparando a versão contaminada e descontaminada. Comparando os resultados das versões contaminada e descontaminada do *sub-corpus*, apresentados na Tabela 9, observa-se que a maioria dos modelos, exceto o RoBERTaLexPT, apresentou uma redução de aproximadamente 1% no F1-Score. Contudo, todos os mode-

los exibiram valores de p acima de 0,05, indicando que não houve diferença significativa no desempenho entre as versões com e sem contaminação. Os resultados com as demais métricas detalhadas de cada versão do *corpus* podem sem encontradas nas Tabelas 7 e 8.

A ausência de significância estatística na diferença dos resultados dos modelos entre as versões contaminada e descontaminada do *sub-corpus* pode ser atribuída a duas hipóteses: (i) as instâncias removidas não impactavam significativamente o desempenho dos modelos ou (ii) a eliminação do vazamento de instâncias duplicadas, aliada à correção de anotações ambíguas e parciais, gerou um equilíbrio entre a diminuição do F1-Score pela remoção das duplicatas e o aumento do F1-Score pela melhoria na qualidade das anotações, resultando em uma performance praticamente inalterada.

Neste estudo, não avaliamos os resultados em diferentes cenários com variações no tipo de contaminação corrigida. No entanto, esse é um experimento que pode ser conduzido no futuro para compreender melhor o impacto de cada nível de contaminação.

Embora não tenha sido encontrada diferença significativa entre os modelos nas versões contaminada e descontaminada do *sub-corpus*, foi possível observar variações significativas entre os modelos dentro de cada versão, quando comparados entre si. A Figura 2b apresenta uma topologia distinta da versão descontaminada (Figura 2a), evidenciada pela redução de conexões entre os nós e pelo aumento de nós desconectados. Por exemplo, o modelo BERTikal possui três arestas na versão contaminada, enquanto apresenta cinco na versão descontaminada.

Além disso, diferenças estatísticas entre os modelos também variam conforme a versão do *sub-corpus*. Na versão contaminada, os modelos XLM-R (base) e RoBERTaLexPT não apresentam diferenças significativas entre os demais modelos, mas passam a apresentar na versão descontaminada. Da mesma forma, o BERTimbau (*large*) apresenta diferença significativa em relação ao Jurisbert, enquanto o BERTimbau (base) ganha diferença significativa em relação ao LegalBert-pt SC.

Assim, observamos que, apesar da ausência de diferença entre as versões dos modelos, a relação entre os modelos muda. Ou seja, as versões contaminada e descontaminada do *sub-corpus* fornecem diferentes perspectivas sobre o desempenho relativo dos modelos, podendo alterar quais modelos se destacam em cada cenário.

6.2. Análise dos resultados do *sub-corpus* de tipos

Analisando os resultados, encontramos alguns pontos importantes, especialmente quando comparada ao nível de categorias. A Tabela 7 demonstra que, embora o desempenho geral do *sub-corpus* descontaminado tenha caído em relação às categorias, isso é esperado devido à maior complexidade do *sub-corpus* de tipos, que inclui 16 classes especializadas, enquanto o *sub-corpus* de categorias tem apenas 7 classes mais amplas. Essa maior granularidade torna a tarefa de classificação mais desafiadora, justificando a queda nos valores de F1-Score.

Os resultados do *sub-corpus* de tipos revelam uma discrepância semelhante à observada na Subseção 6.1, com o pior modelo alcançando um F1-Score de 63,36% e o melhor, 78,59%, apresentando uma diferença de 15,23 pontos percentuais. Essa variação reforça a disparidade na eficácia dos diferentes modelos de NER, evidenciando que a escolha do modelo impacta significativamente o desempenho na tarefa.

Sobre o melhor modelo, não é possível determinar qual se destaca de forma clara com base apenas nas médias. Dos 9 modelos avaliados, 7 apresentaram médias acima de 70%, com a maior sendo 78,59% e outros cinco modelos variando entre 76,31% e 79,79%. Dessa forma, no nível de tipos, também não é possível concluir que modelos específicos para o domínio legislativo ou jurídico apresentam desempenho superior, nem que modelos desenvolvidos exclusivamente para o português superam os multilíngues em termos de resultados.

Em relação ao desvio padrão, o BERTimbau (versão *large*) novamente se destaca por apresentar o maior desvio padrão entre os melhores modelos, assim como observado no nível de categorias. Neste caso, ele alcança o maior desvio padrão dentre todos os modelos avaliados. Por outro lado, o XLM-R, excetuando o BERTikal, que apresentou o pior desempenho geral, registrou o menor desvio padrão, consolidando-se como uma alternativa sólida para a tarefa, ao aliar resultados competitivos à menor variância. Essa arquitetura tem se mostrado competitiva com o BERTimbau no domínio legislativo, conforme também observado no nível de categorias (veja a Seção 6.1), especialmente em sua versão adaptada para textos legislativos em português (RoBERTaLexPT).

Modelo	Precisão	Revocação	F1
BERTimbau	75,71 ± 4,82	81,75 ± 5,16	78,59 ± 4,83
XLM-R-base	73,25 ± 2,99	80,73 ± 2,18	76,79 ± 2,43
BERTimbau-base	73,34 ± 3,18	80,26 ± 2,44	76,63 ± 2,71
LegalBert-pt_FP	73,05 ± 3,50	80,52 ± 2,83	76,59 ± 3,03
RoBERTaLexPT	73,38 ± 3,62	80,15 ± 2,83	76,59 ± 3,03
BERTimbaulaw	72,69 ± 3,01	80,34 ± 2,33	76,31 ± 2,53
Jurisbert	70,52 ± 3,71	78,11 ± 3,30	74,09 ± 3,30
LegalBert-pt_SC	64,94 ± 3,70	72,41 ± 3,60	68,43 ± 3,23
BERTikal	61,13 ± 1,97	65,79 ± 2,19	63,36 ± 1,78

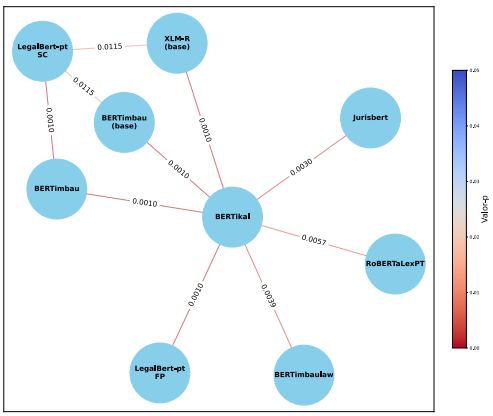
Tabela 10: Resultados da avaliação dos modelos no ***sub-corpus*** **descontaminado de tipos**.

Modelo	Precisão	Revocação	F1
BERTimbau-base	81,03 ± 03,37	86,57 ± 02,41	83,70 ± 02,82
RoBERTaLexPT	80,82 ± 02,95	85,96 ± 02,25	83,30 ± 02,48
BERTimbaulaw	80,38 ± 02,38	86,09 ± 01,83	83,13 ± 01,92
LegalBert-pt_FP	80,01 ± 03,11	86,07 ± 02,21	82,92 ± 02,50
XLM-R-base	80,09 ± 03,08	85,95 ± 02,41	82,91 ± 02,65
Jurisbert	78,47 ± 02,98	84,81 ± 02,59	81,51 ± 02,65
LegalBert-pt_SC	75,76 ± 03,80	82,09 ± 03,18	78,78 ± 03,31
BERTimbau	76.12 ± 07.86	75,03 ± 07,34	74.64 ± 07.59
BERTikal	68,73 ± 02,02	74,65 ± 02,07	71,56 ± 01,85

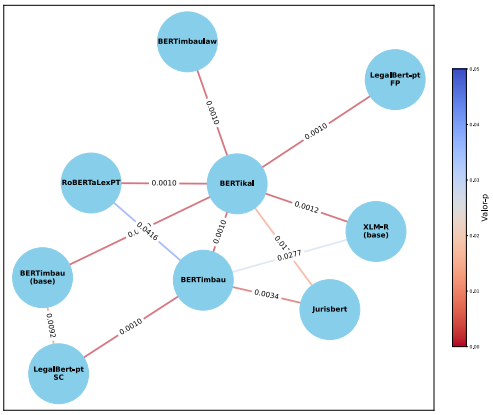
Tabela 11: Resultados da avaliação dos modelos no ***sub-corpus*** **contaminado de tipos**.

Modelo	F1 (Contaminado)	F1 (Descontaminado)	Δ F1	Valor-p
BERTimbau	74.64 ± 07.59	78,59 ± 04,83	+03.36	0,000031
XLM-R-base	82,91 ± 02,65	76,79 ± 02,43	-06,12	0,000031
BERTimbau-base	83,70 ± 02,82	76,63 ± 02,71	-07,07	0,000031
LegalBert-pt-FP	82,92 ± 02,50	76,59 ± 03,03	-06,33	0,000031
RoBERTaLexPT	83,30 ± 02,48	76,59 ± 03,03	-06,71	0,000031
BERTimbau-law	83,13 ± 01,92	76,31 ± 02,53	-06,82	0,000031
Jurisbert	81,51 ± 02,65	74,09 ± 03,30	-07,42	0,000031
LegalBert-pt-SC	78,78 ± 03,31	68,43 ± 03,23	-10,35	0,000031
BERTikal	71,56 ± 01,85	63,36 ± 01,78	-08,20	0,001474

Tabela 12: Resultados da avaliação dos modelos no ***sub-corpus*** **de tipos** com e sem contaminação, incluindo a diferença na métrica F1 e o valor-p obtido pelo teste de Wilcoxon Signed-Rank.



(a) Versão descontaminada.



(b) Versão contaminada.

Figura 3: Gráficos de rede que ilustram a comparação entre os modelos, utilizando o ***sub-corpus* de tipos**. As arestas conectam os modelos que apresentam diferenças significativas, sendo o valor-p mapeado para cor da aresta. Nodos desconectados não tem diferenças significativas para os demais. As figuras com os resultados completos estão disponíveis no Apêndice A.

Em relação à comparação do desempenho dos modelos aplicados à versão descontaminada do *subcorpus* de tipos, a Figura 3a apresenta as diferenças significativas entre os modelos, conforme os resultados do teste de Friedman seguido do *post-hoc* de Nemenyi. Observa-se que o modelo BERTikal, como esperado, exibe diferenças significativas em relação a todos os modelos com F1-Score acima de 70%, exceto o LegalBert-pt-SC, cujo desempenho é mais próximo ao seu. Além disso, os modelos BERTimbau (em ambas as versões) e XLM-R também apresentam diferenças significativas em relação ao LegalBert-pt-SC, que possui o segundo pior desempenho. Esses resultados sugerem uma tendência para a escolha dos modelos BERTimbau e XLM-R, dado que se destacam com o maior número de diferenças estatísticas e, simultaneamente, apresentam os melhores desempenhos gerais.

Comparando a versão contaminada e descontaminada. A Tabela 12 mostra a diferença dos F1-Scores entre a versão contaminada e descontaminada do *sub-corpus* com os modelos avaliados (demais completas podem ser encontradas nas Tabelas 10 e 11). Em relação ao já analisado no *sub-corpus* de categorias, chama a atenção dois pontos discrepantes: maiores diferenças e significância estatística na comparação de todos os modelos.

Tanto as diferenças quanto a significância estatística podem ser atribuídas à natureza dos *sub-corpus*. O nível de tipos apresenta uma maior complexidade de classificação devido ao maior número de classes e às especificações mais detalhadas em comparação ao nível de categorias (veja as Tabelas 1 e 6). Além disso, classes derivadas de uma mesma categoria podem compartilhar similaridades semânticas, dificultando a distinção entre elas pelos modelos. Outro fator relevante é o menor número de exemplos disponíveis no *sub-corpus* de tipos em relação ao de categorias, além de um menor número médio de exemplos por classe, o que acentua o desafio de treinamento para os modelos.

O BERTimbau (versão *large*), apesar de alcançar a maior média no *sub-corpus* descontaminado, apresentou uma média de 70,03% com desvio padrão de 7,34 no *sub-corpus* contaminado, evidenciando um desempenho inconsistente entre as duas versões do *sub-corpus*. Esse elevado desvio padrão indica uma alta variabilidade nos resultados ao longo das execuções, o que reflete um comportamento instável em relação aos demais modelos avaliados. Essa variação pode ser associada à presença de vazamento de informações no *sub-corpus* contaminado, que tende a mascarar a real capacidade de generalização do modelo.

Esse comportamento impactou diretamente a análise estatística entre os modelos no *sub-corpus* contaminado (veja Figura 3b). Enquanto no *sub-corpus* descontaminado o BERTimbau apresentou diferenças estatisticamente significativas apenas com os dois piores modelos, no contaminado ele mostrou diferenças significativas com cinco modelos, incluindo o RoBERTaLexPT e o XLM-R, que estão entre os melhores e destacados previamente. Curiosamente, uma exceção foi sua própria versão base, o BERTimbau-base, que obteve a melhor média no *sub-corpus* contaminado. Esses resultados reforçam o impacto negativo do vazamento de informações, não apenas no desempenho médio, mas também na estabilidade observada, destacando a importância de *sub-corpus* descontaminados para uma análise mais confiável e robusta.

Modelo	Garcia et al. (2024)	Contaminado	Descontaminado	Δ C	Δ D
BERTimbau-base	86,39	83,46 ± 1,89	82,49 ± 1,92	-02,96	-03,90
BERTimbau	87,77	84,21 ± 2,06	83,42 ± 2,50	-03,56	-04,35
BERTikal	79,21	71,83 ± 2,46	69,59 ± 2,78	-07,38	-09,62
JurisBert	81,67	79,52 ± 2,16	78,48 ± 2,34	-02,15	-03,19
BERTimbaulaw	87,11	81,97 ± 2,17	81,62 ± 2,10	-05,14	-05,49
RoBERTaLexPT	88,56	82,21 ± 2,47	82,51 ± 1,83	-06,35	-06,07

Tabela 13: Comparação do F1-Score macro para cada modelo no **nível de categorias** entre os experimentos de Garcia et al. (2024) e os nossos experimentos. Δ *C* faz referência à diferença do resultado de (Garcia et al., 2024) e os resultados do *sub-corpus* contaminado, enquanto Δ *D* faz referência à diferença com os resultados do *sub-corpus* descontaminado.

6.3. Comparando com a literatura

Conforme discutido na Seção 3.2, a presença de instâncias duplicadas pode provocar *overfitting* e inflar os resultados das métricas de avaliação. Esse comportamento foi observado quando comparamos nossos resultados com os da literatura (Albuquerque et al., 2023; Nunes et al., 2024b; Garcia et al., 2024).

Albuquerque et al. (2023) e Nunes et al. (2024b) avaliaram o *sub-corpus* de categorias utilizando técnicas de avaliação baseadas em cálculos de média e desvio padrão, visando obter uma análise mais robusta dos resultados. Ambos os estudos realizaram cinco execuções dos modelos para calcular essas métricas, com a principal diferença sendo que Albuquerque et al. (2023) utilizou divisões aleatórias dos dados, enquanto Nunes et al. (2024b) adotou validação cruzada com cinco partições estratificadas.

Em Albuquerque et al. (2023), foi utilizado um modelo[9] baseado no BERTimbau base, que teve seu pré-treinamento continuado com o *corpus* LeNER-Br (Luz de Araujo et al., 2018), seguido de *finetuning* no mesmo *corpus*. Já Nunes et al. (2024b) empregou o BERTimbau base, sem pré-treinamento adicional em *corpus* legal. Ambos os estudos utilizaram uma camada linear para classificar as entidades nomeadas.

Nos dois estudos, os F1-Scores macro ficaram próximos de 83%, sendo 83,17 ± 1,15 em Albuquerque et al. (2023) e 83,53 ± 2,56 em Nunes et al. (2024b). Apesar de pequenas diferenças nas médias e nos desvios padrão, os resultados demonstram uma proximidade significativa.

Garcia et al. (2024) também avaliou o BERTimbau base, mas utilizando a estratégia de *holdout* com divisão aleatória dos dados, conforme a separação disponibilizada no artigo do *corpus* (Albuquerque et al., 2022). Nesse caso, foi obtido um F1-Score macro de 86,39%, aproximadamente 3% superior aos resultados reportados por Albuquerque et al. (2023) e Nunes et al. (2024b). Essa diferença pode ser atribuída à execução única do modelo, que pode não refletir com precisão o desempenho real, ou à seleção aleatória de um conjunto de teste mais fácil, o que pode ter proporcionado um resultado otimista.

Comparando os resultados com os obtidos neste trabalho, observa-se uma pequena diferença entre eles (Albuquerque et al., 2023; Nunes et al., 2024b), no qual o resultado obtido neste trabalho com o retreinamento do modelo contamiado obteve média e desvio padrão entre os dois anteriores. Entretanto, encontramos um desempenho aproximadamente 3% inferior quando comparado ao estudo que utilizou apenas uma execução (Garcia et al., 2024). Além disto, refizemos o *finetuning* dos demais modelos testados por Garcia et al. (2024). A única exceção foi o Albertina e versões maiores dos modelos XLM-R que por limitações de hardware não puderam ser executadas.

Os resultados dos experimentos realizados no *sub-corpus* de categorias, comparando os valores reportados por Garcia et al. (2024) com os obtidos com o *corpus* atualizado, podem ser vistos na Tabela 13. Além do decréscimo já mencionado para o BERTimbau base, observa-se que os demais modelos também apresentaram quedas de desempenho, com o Jurisbert mostrando o menor decréscimo (-2,15) e o BERTikal o maior (-7,38).

Ao compararmos com os resultado sobre o *sub-corpus* descontaminado, encontramos uma maior diferença com os trabalhos anteriores. Primeiro, temos um decréscimo de cerca de 1% em relação aos trabalhos de Albuquerque et al. (2023) e Nunes et al. (2024b). Segundo, as variações comparadas com os testes usando *holdout* de Garcia et al. (2024) se tornam maiores, vari-

[9] https://huggingface.co/pierreguillou/ner-bert-base-cased-pt-lenerbr

Modelo	Garcia et al. (2024)	Contaminado	Descontaminado	Δ C	Δ D
BERTimbau	84,74	74.64 ± 07.59	78,59 ± 04,83	-10,10	-06,15
BERTimbau-base	83,83	83,70 ± 02,82	76,63 ± 02,71	-00,13	-07,20
BERTikal	75,70	71,56 ± 01,85	63,36 ± 01,78	-04,14	-12,34
JurisBert	77,97	81,51 ± 02,65	74,09 ± 03,30	+03,54	-03,88
BERTimbaulaw	84,42	83,13 ± 01,92	76,31 ± 02,53	-01,29	-08,11
RoBERTaLexPT	86,03	83,30 ± 02,48	76,59 ± 03,03	-02,73	-09,44

Tabela 14: Comparação do F1-Score macro para cada modelo no **nível de tipos** entre os experimentos de Garcia et al. (2024) e os nossos experimentos. Δ *C* faz referência a diferência do resultado de (Garcia et al., 2024) e os resultados do *sub-corpus* contaminado, enquanto Δ *D* faz referência a diferência com os resultados do *sub-corpus* descontaminado.

ando entre 3,19% e 9,62%. Os decréscimos apresentados em ambas as versões do *sub-corpus* reinteram a importância de múltiplas execuções de modelos para se obter uma faixa de resultados mais confiável.

Embora os modelos BERTimbau e RoBERTaLexPT tenham alcançado as maiores médias, os testes estatísticos (ver Subseção 6.1) indicam que não há uma diferença significativa entre eles e os demais modelos testados por Garcia et al. (2024), com poucas exceções, como o modelo BERTikal. No estudo original, os autores não realizaram testes estatísticos; contudo, dada a proximidade dos resultados, é provável que não haja uma diferença significativa entre o RoBERTaLex (que apresentou o melhor desempenho) e os demais modelos. Essa situação é semelhante à observada em nosso trabalho, reforçando a ideia de que o desempenho entre os modelos é bastante próximo, excetuando-se *outliers* como o BERTikal.

O mesmo se aplica ao *sub-corpus* de tipos, onde identificamos apenas experimentos utilizando modelos *transformers* no trabalho de Garcia et al. (2024). A comparação entre os resultados desse estudo e os nossos pode ser vista na Tabela 14. Nela, observamos uma diferença menos acentuada entre os resultados obtidos com o *sub-corpus* original e aqueles com o *sub-corpus* contaminado em nosso experimentos, sendo o BERTimbau base o modelo com o menor decréscimo de desempenho (-00,13) e o BERTikal apresentando o maior decréscimo (-04,14), ademais houve um acréscimo no resultado do JurisBert de +03,54 pontos percentuais. Este acréscimo reforça a importância da utilização de métodos que testam o modelo repetidamente, como a validação cruzada, visto que métodos como *holdout* podem não somente proporcionar resultados maiores que o esperado, mas também menores, tal como visto no exemplo do JurisBert.

Ao compararmos os resultados de (Garcia et al., 2024) nas versões contaminada e descontaminada do *sub-corpus* no nível de tipos, observamos uma variação significativamente maior do que a registrada no nível de categorias (veja Tabela 13). Uma possível explicação para essa maior variação é o menor número de instâncias no nível de tipos (3.006 entidades em 2.424 sentenças) em comparação ao nível de categorias (3.537 entidades em 2.468 sentenças). Além disso, o nível de tipos possui um número maior de classes (16) em relação ao nível de categorias (7), o que torna a tarefa de classificação mais complexa. Essa combinação de menor quantidade de dados e maior número de classes amplifica o impacto da eliminação de vazamento, resultando em maior variação nos resultados.

6.4. Utilização de autoaprendizagem

A seleção do valor do *threshold* é um aspecto crítico de nossa abordagem, pois desempenha um papel fundamental na determinação do desempenho geral do sistema de NER. Na fase de validação cruzada, nossa avaliação revelou que o valor do *threshold* de 0,99 demonstrou consistentemente desempenho superior, resultando em um F1-Score na faixa de 86,70 ± 2,28 em todos as partições. É importante destacar que *thresholds* de 0,95 e 0,975 apresentam resultados semelhantes, conforme mostrado na Tabela 15. Assim, para escolher o *threshold* entre eles, selecionamos aquele que apresenta maior ganho na maioria das entidades. Baseamos essa escolha no melhor F1-Score obtido na validação cruzada, conforme detalhado na Seção 5.5, o que provou ser a decisão mais eficaz nos cinco folds. Consequentemente, os resultados apresentados nesta seção abrangem as métricas finais obtidas com o *threshold* definido em 0,99.

A Tabela 15 mostra o impacto do aprendizado automático nos resultados finais. Esta tabela apresenta o F1-Score para as classes de entidades. Nossa abordagem foi capaz de alcançar resultados significativamente superiores para a maioria das classes, como pode ser observado na entidade "LOCAL", que teve um aumento de 8% em seu F1-Score. De forma semelhante, "ORGANIZACAO" apresentou um aumento de 6% e uma redução no desvio padrão, assim como "PRODUTODELEI", que teve um incremento de 5% e também uma redução no desvio padrão. É interessante destacar a entidade "EVENTO", que não pôde ser prevista com os dados originais, apresentando um F1-Score de 0,0 ± 0,0, e conseguimos alcançar 58,10 ± 34,16. As entidades "DATA", "FUNDAMENTO" e "PESSOA" não apresentaram impactos significativos, registrando apenas pequenos aumentos na média ou reduções no desvio padrão.

No artigo original do UlyssesNER-Br (Albuquerque et al., 2022), os autores utilizaram o *corpus* para treinar um modelo de HMM e um modelo de CRF. Além disso, também utilizaram a arquitetura BiLSTM-CRF e a arquitetura Glove para comparar com os resultados obtidos no trabalho de Luz de Araujo et al. (2018) com o *corpus* LeNER-Br. Nesse sentido, a Tabela 16 demonstra os resultados superiores, nos quais apenas o uso de um modelo BERT ajustado para o português (Souza et al., 2020) conseguiu aumentar o F1-Score em 6,64%. No entanto, a introdução do aprendizado automático se mostrou um fator importante no aumento do F1-Score em 9,81%.

Para realizar uma análise mais aprofundada dos resultados, realizamos um treinamento final com os dados utilizados na validação cruzada e os testamos em um conjunto de dados previamente não utilizado. A abordagem de validação cruzada com divisão serviu a um duplo propósito: ela não apenas ajudou no ajuste do hiperparâmetro do *threshold*, mas também ofereceu uma maneira mais abrangente de validar os resultados com um número predefinido de divisões. Além disso, essa abordagem possibilitou uma análise detalhada dos resultados no conjunto de teste, com atenção especial ao impacto e às consequências de cada categoria.

A curva de aprendizado para cada entidade, conforme ilustrado na Figura 4, mostra o impacto significativo do autoaprendizado nas classes, resultando em um aumento considerável na métrica em comparação com o resultado padrão na iteração zero. No entanto, vale ressaltar que algumas oscilações foram observadas em iterações específicas, possivelmente devido à introdução de

Threshold	DATA	EVENTO	FUNDAMENTO	LOCAL	ORGANIZACAO	PESSOA	PRODUTODELEI	F1 macro
0,9	94,49 ± 03,13	48,89 ± 33,41	88,01 ± 02,12	85,54 ± 03,50	82,56 ± 05,95	83,98 ± 04,06	75,11 ± 05,72	85,02 ± 02,45
0,925	94,62 ± 02,65	54,53 ± 32,68	89,34 ± 01,61	85,10 ± 04,34	83,16 ± 04,49	84,40 ± 03,55	71,36 ± 04,14	85,12 ± 02,31
0,95	**95,08 ± 01,89**	49,05 ± 33,86	89,99 ± 02,45	87,49 ± 05,03	**84,82 ± 02,57**	85,55 ± 05,24	76,01 ± 07,18	**86,56 ± 1,99**
0,975	95,08 ± 03,41	50,48 ± 34,02	**90,50 ± 02,54**	85,11 ± 04,25	85,30 ± 05,75	85,75 ± 04,83	75,83 ± 05,94	**86,48 ± 02,91**
0,99	94,77 ± 02,65	**58,10 ± 34,16**	88,60 ± 02,29	**86,46 ± 03,73**	84,89 ± 05,77	**87,48 ± 02,79**	**75,42 ± 04,47**	**86,70 ± 02,28**
0,9975	93,35 ± 02,66	03,64 ± 07,27	88,91 ± 02,24	80,78 ± 02,89	79,12 ± 04,33	87,91 ± 03,32	72,81 ± 05,65	84,16 ± 02,28
0,999	93,10 ± 02,21	20,00 ± 24,49	88,37 ± 01,65	80,87 ± 04,06	79,73 ± 02,94	87,22 ± 03,02	70,74 ± 09,06	83,87 ± 02,36
Standard	94,25 ± 02,69	00,00 ± 00,00	88,59 ± 03,86	78,96 ± 03,90	78,33 ± 04,44	87,77 ± 03,19	70,44 ± 07,40	83,53 ± 02,56

Tabela 15: Resultados da validação cruzada para cada *threshold* com autoaprendizagem e o resultado sem autoaprendizagem.

Modelo	Acurácia	Precisão	Revocação	F1-Score
HMM	93,07 ± 00,78	60,45 ± 02,18	30,82 ± 01,81	40,74 ± 01,83
CRF	97,27 ± 00,77	83,42 ± 00,91	70,40 ± 01,54	76,28 ± 01,12
BiLSTM-CRF + Glove	97,66 ± 00,47	80,48 ± 02,69	73,63 ± 02,65	76,89 ± 02,49
BERTimbau	98,30 ± 00,32	80,17 ± 03,67	87,63 ± 01,13	83,53 ± 02,56
BERTimbau + Autoaprendizagem	**98,45 ± 00,24**	**85,37 ± 02,91**	**89,02 ± 01,45**	**86,70 ± 02,28**

Tabela 16: Resultados originais e nossos resultados com BERT e *self-learning* utilizando o *threshold* de 0,99.

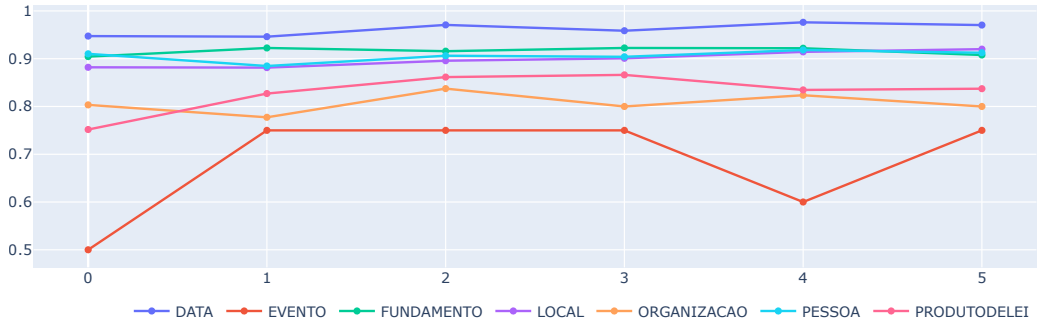


Figura 4: F1-Score para cada entidade ao longo das iterações.

exemplos mal anotados no conjunto de treinamento. Apesar disso, a tendência geral demonstra a robustez do uso do autoaprendizado e destaca a influência do *threshold* escolhido na filtragem de uma parte substancial dos dados ruidosos.

Da mesma forma, a Figura 5a ilustra a curva de aprendizado para o número acumulado de sentenças adicionadas ao longo das iterações, com foco no F1-Score geral. Na quarta iteração, alcançamos nosso maior F1-Score de 90%, destacando o impacto positivo do aumento do *corpus* original. Esse resultado é bastante promissor em comparação com o F1-Score de 87,28% obtido usando apenas o modelo BERT na iteração zero.

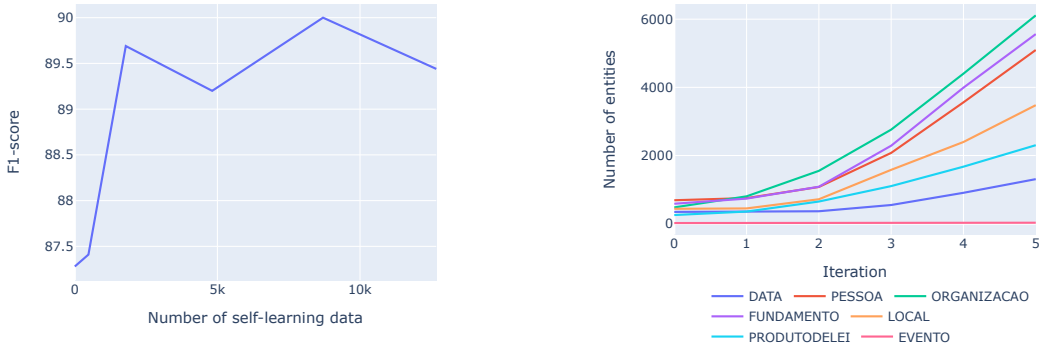
A Figura 5b mostra o aumento no número de exemplos para cada entidade durante as iterações. Vale ressaltar que tanto as classes com mais quanto as com menos dados apresentaram um aumento considerável no número de exemplos. Mesmo assim, as classes "DATA" e "EVENTO" tiveram o menor aumento. Acreditamos que esse fato ocorreu porque as datas possuem formatos específicos, facilitando a filtragem de ruído, e "EVENTO", por ser a classe minoritária, aumentou ligeiramente ao longo das iterações justamente devido ao seu pequeno número de dados.

7. Conclusão

Neste trabalho, abordamos de maneira abrangente o problema do vazamento de dados no treinamento de modelos de Reconhecimento de Entidades Nomeadas (NER) em textos legislativos em português brasileiro. Identificamos como duplicatas e anotações inconsistentes comprometem a avaliação dos modelos, levando a resultados superestimados. Através de um processo cuidadoso de correção, removemos duplicatas e padronizamos as anotações, resultando em um *corpus* de maior qualidade, o que nos permitiu realizar um novo *benchmark*.

Essas melhorias no tratamento dos dados proporcionaram uma avaliação mais precisa dos modelos e destacaram a importância de uma abordagem rigorosa no pré-processamento de *corpora* para tarefas de Processamento de Linguagem Natural (PLN). A pesquisa ressalta que o vazamento de dados é um aspecto relevante que merece atenção para assegurar a validade dos resultados em contextos especializados.

Ademais, apresentamos um método de NER utilizando autoaprendizagem e amostragem ativa, aplicando-o ao texto legislativo em português do *corpus* UlyssesNER-BR. Nossos resultados mostram que o BERTimbau com autoaprendizagem obteve uma F1-score média geral de 86,70 ± 2,28 na validação cruzada e um resultado final de 90%, demonstrando um desempenho robusto no reconhecimento de entidades em comparação ao uso apenas do BERTimbau e aos *benchmarks* anteriores. Esse achado destaca a eficácia do BERTimbau com autoaprendizagem para o Reconhecimento de Entidades Nomeadas no domínio legal/legislativo, evidenciando seu potencial para tarefas de análise de texto jurídico.



(a) Curva de aprendizado mostrando a relação entre o número cumulativo de dados adicionados em cada iteração e seu respectivo F1-Score.

(b) Número cumulativo de exemplos de cada entidade adicionados em cada iteração.

Figura 5: Impacto da autoaprendizagem na fase de treinamento.

Apesar dos resultados positivos, nosso estudo em autoaprendizagem apresenta algumas limitações. Realizamos os experimentos apenas no nível de categoria de entidade, sem avaliar o comportamento do modelo no nível de tipo, nem o impacto da contaminação na autoaprendizagem. Como trabalho futuro, planejamos explorar o nível de tipo e as correlações entre os níveis, além de investigar o efeito da contaminação. Também pretendemos adotar uma abordagem de ensemble do nosso modelo com o BERTimbau afinado com o *corpus* LeNER-Br[10] utilizando entidades equivalentes entre os *corpora* no ensemble. Em relação ao ajuste fino dos modelos, planejamos realizar experimentos com as versões BERT-CRF e BERT-LSTM-CRF do BERTimbau, disponíveis no repositório oficial[11]. Também pretendemos realizar experimentos com outros modelos recentes baseados em BERT para o português, como o Albertina (Rodrigues et al., 2023). Nossos resultados também destacam a importância de um conjunto de dados diverso e representativo para o ajuste fino de modelos em domínios específicos. Pesquisas futuras devem focar na ampliação dos dados de treinamento, na curadoria de novos dados com especialistas para torná-los disponíveis para uso geral e na exploração de outras técnicas de pré-treinamento e ajuste fino para melhorar o desempenho de modelos de NER no domínio legislativo.

[10] https://huggingface.co/Luciano/bertimbau-large-lener_br

[11] https://github.com/neuralmind-ai/portuguese-bert/tree/master/ner_evaluation

Agradecimentos

Este trabalho foi parcialmente financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Também agradecemos o apoio financeiro da agência de fomento brasileira CNPq. Alguns experimentos neste trabalho utilizaram a infraestrutura PCAD, http://gppd-hpc.inf.ufrgs.br, no INF/UFRGS.

Referências

Albuquerque, Hidelberg O., Rosimeire Costa, Gabriel Silvestre, Ellen Souza, Nádia F. F. da Silva, Douglas Vitório, Gyovana Moriyama, Lucas Martins, Luiza Soezima, Augusto Nunes, Felipe Siqueira, João P. Tarrega, Joao V. Beinotti, Marcio Dias, Matheus Silva, Miguel Gardini, Vinicius Silva, André C. P. L. F. de Carvalho & Adriano L. I. Oliveira. 2022. UlyssesNER-Br: a corpus of brazilian legislative documents for named entity recognition. Em *Computational Processing of the Portuguese Language (PROPOR)*, 3–14. 10.1007/978-3-030-98305-5_1

Albuquerque, Hidelberg O., Ellen Souza, Adriano L. I. Oliveira, David Macêdo, Cleber Zanchettin, Douglas Vitório, Nádia F. F. da Silva & André C. P. L. F. de Carvalho. 2023. On the assessment of deep learning models for named entity recognition of brazilian legal documents. Em *EPIA Conference on Artificial Intelligence*, 93–104. 10.1007/978-3-031-49011-8_8

Alves-Pinto, Ana, Christoph Demus, Michael Spranger, Dirk Labudde & Eleanor Hobley. 2021. Iterative named entity recognition with conditional random fields. *Applied Sciences* 12(1). 330. 10.3390/app12010330

Anaby-Tavor, Ateret, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper & Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! Em *AAAI Conference on Artificial Intelligence*, vol. 34 05, 7383–7390. 10.1609/aaai.v34i05.6233

Angelidis, Iosif, Ilias Chalkidis & Manolis Koubarakis. 2018. Named entity recognition, linking and generation for greek legislation. Em *International Conference on Legal Knowledge and Information Systems (JURIX)*, 1–10. 10.3233/978-1-61499-935-5-1

Badji, Ines. 2018. *Legal entity extraction with NER systems*: Universidad Politécnica de Madrid. Tese de Doutoramento.

Balloccu, Simone, Patrícia Schmidtová, Mateusz Lango & Ondrej Dusek. 2024. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs. Em *18th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 67–93.

Bird, Steven. 2006. NLTK: the natural language toolkit. Em *21st International Conference on Computational Linguistics (COLING) and 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, 69–72. 10.3115/1225403.1225421

Bonifacio, Luiz Henrique, Paulo Arantes Vilela, Gustavo Rocha Lobato & Eraldo Rezende Fernandes. 2020. A study on the impact of intradomain finetuning of deep language models for legal named entity recognition in Portuguese. Em *Intelligent Systems*, 648–662. 10.1007/978-3-030-61377-8_46

Brito, Maurício, Vládia Pinheiro, Vasco Furtado, João Araújo Monteiro Neto, Francisco das Chagas Jucá Bomfim, André Câmara Ferreira da Costa & Raquel Silveira. 2023. CDJUR-BR: Uma coleção dourada do judiciário Brasileiro com entidades nomeadas refinadas. Em *14th Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, 177–186. 10.5753/stil.2023.234217

Chen, Yukun, Thomas A. Lasko, Qiaozhu Mei, Joshua C. Denny & Hua Xu. 2015. A study of active learning methods for named entity recognition in clinical text. *Journal of Biomedical Informatics* 58. 11–18. 10.1016/j.jbi.2015.09.010

Clark, Kevin, Minh-Thang Luong, Christopher D. Manning & Quoc V. Le. 2018. Semi-supervised sequence modeling with cross-view training. arXiv [cs.CL]. 10.48550/arXiv.1809.08370

Cohen, Aaron M. 2005. A survey of current work in biomedical text mining. *Briefings in Bioinformatics* 6(1). 57–71. 10.1093/bib/6.1.57

Collovini, Sandra, Joaquim Francisco Santos Neto, Bernardo Scapini Consoli, Juliano Terra, Renata Vieira, Paulo Quaresma, Marlo Souza, Daniela Barreiro Claro & Rafael Glauber. 2019. IberLEF 2019 Portuguese named entity recognition and relation extraction tasks. Em *Iberian Languages Evaluation Forum (IberLEF)*, 390–410.

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer & Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. Em *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer & Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. arXiv [cs.CL]. 10.48550/arXiv.1911.02116

Correia, Fernando A., Alexandre A.A. Almeida, José Luiz Nunes, Kaline G. Santos, Ivar A. Hartmann, Felipe A. Silva & Hélio Lopes. 2022. Fine-grained legal entity annotation: A case study on the Brazilian supreme court. *Information Processing & Management* 59(1). 102794. 10.1016/j.ipm.2021.102794

Darji, Harshil, Jelena Mitrović & Michael Granitzer. 2023. German BERT model for legal named entity recognition. arXiv [cs.CL/cs.LG]. 10.48550/arXiv.2303.05388

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv [cs.CL]. arXiv.1810.04805

Dong, Xin Luna & Gerard de Melo. 2019. A robust self-learning framework for cross-lingual

text classification. Em *Conference on Empirical Methods (EMNLP) in Natural Language Processing and International Joint Conference on Natural Language Processing (IJCNLP)*, 6306–6310. 10.18653/v1/D19-1658

Dozier, Christopher, Ravikumar Kondadadi, Marc Light, Arun Vachher, Sriharsha Veeramachaneni & Ramdev Wudali. 2010. *Named entity recognition and resolution in legal text* 27–43. Springer Berlin Heidelberg. 10.1007/978-3-642-12837-0_2

Dupre, Robert, Jiri Fajtl, Vasileios Argyriou & Paolo Remagnino. 2019. Improving dataset volumes and model accuracy with semi-supervised iterative self-learning. *IEEE Transactions on Image Processing* 29. 4337–4348. 10.1109/tip.2019.2913986

Feng, Steven Y., Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura & Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. arXiv [cs.CL]. 10.48550/arXiv.2105.03075

Friedman, Milton. 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association* 32(200). 675–701. 10.2307/2279372

Gao, Shang, Olivera Kotevska, Alexandre Sorokine & J Blair Christian. 2021. A pre-training and self-training approach for biomedical named entity recognition. *PLOS ONE* 16(2). e0246310. 10.1371/journal.pone.0246310

Garcia, Eduardo AS, Nadia FF Silva, Felipe Siqueira, Hidelberg O Albuquerque, Juliana RS Gomes, Ellen Souza & Eliomar A Lima. 2024. RoBERTaLexPT: A legal RoBERTa model pretrained with deduplication for Portuguese. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 374–383.

Glaser, Ingo, Bernhard Waltl & Florian Matthes. 2018. Named entity recognition, extraction, and linking in german legal contracts. Em *IRIS: Internationales Rechtsinformatik Symposium*, 325–334

Gomes, Diogo da Silva Magalhães, Fábio Corrêa Cordeiro, Bernardo Scapini Consoli, Nikolas Lacerda Santos, Viviane Pereira Moreira, Renata Vieira, Silvia Moraes & Alexandre Gonçalves Evsukoff. 2021. Portuguese word embeddings for the oil and gas industry: Development and evaluation. *Computers in Industry* 124. 103347. 10.1016/j.compind.2020.103347

He, Pengcheng, Xiaodong Liu, Jianfeng Gao & Weizhu Chen. 2021. DeBERTa: decoding-enhanced BERT with disentangled attention. Em *International Conference on Learning Representations (ICLR)*,

Heald, David Albert. 2006. Varieties of transparency. Em Christopher Hood & David Heald (eds.), *Transparency: The Key to Better Governance?*, 25–43. Oxford University Press

Helwe, Chadi & Shady Elbassuoni. 2019. Arabic named entity recognition via deep co-learning. *Artificial Intelligence Review* 52(1). 197–215. 10.1007/s10462-019-09688-6

Kalamkar, Prathamesh, Astha Agarwal, Aman Tiwari, Smita Gupta, Saurabh Karn & Vivek Raghavan. 2022. Named entity recognition in Indian court judgments. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2211.03442

Kim, Ji-Hyun. 2009. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis* 53(11). 3735–3745. 10.1016/j.csda.2009.04.009

Lample, Guillaume, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami & Chris Dyer. 2016. Neural architectures for named entity recognition. arXiv [cs.CL]. 10.48550/arXiv.1603.01360

Lathrop, Daniel & Laurel Ruma. 2010. *Open government: Collaboration, transparency, and participation in practice*. O'Reilly

Leitner, Elena, Georg Rehm & Julian Moreno-Schneider. 2019. Fine-grained named entity recognition in legal documents. Em *15th International Conference on Semantic Systems (SEMANTiCS)*, 272–287. 10.1007/978-3-030-33220-4_20

Li, Bohan, Yutai Hou & Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open* 3. 71–90. 10.1016/j.aiopen.2022.03.001

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer & Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. arXiv [cs.CL]. 10.48550/arXiv.1907.11692

Luz de Araujo, Pedro H., Teófilo E. de Campos, Renato R. R. de Oliveira, Matheus

Stauffer, Samuel Couto & Paulo Bermejo. 2018. LeNER-Br: a dataset for named entity recognition in Brazilian legal text. Em *Computational Processing of the Portuguese Language (PROPOR)*, 313–323. 10.1007/978-3-319-99722-3_32

Machado, Gustavo, Mariana Recamonde Mendoza & Luis Gustavo Corbellini. 2015. What variables are important in predicting bovine viral diarrhea virus? a random forest approach. *Veterinary research* 46. 85. 10.1186/s13567-015-0219-7

Mekala, Dheeraj & Jingbo Shang. 2020. Contextualized weak supervision for text classification. Em *58th Meeting of the Association for Computational Linguistics (ACL)*, 323–333. 10.18653/v1/2020.acl-main.30

Meng, Yu, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang & Jiawei Han. 2020. Text classification using label names only: A language model self-training approach. arXiv [cs.CL/cs.LG]. 10.48550/arXiv.2010.07245

Nemenyi, Peter Bjorn. 1963. *Distribution-free multiple comparisons*: Princeton University. Tese de Doutoramento

Neto, José Reinaldo C. S. A. V. S. & Thiago de Paulo Faleiros. 2021. Deep active-self learning applied to named entity recognition. Em *10th Brazilian Conference on Intelligent Systems (BRACIS)*, 405–418. 10.1007/978-3-030-91699-2_28

Niklaus, Joel, Veton Matoshi, Matthias Stürmer, Ilias Chalkidis & Daniel Ho. 2024. MultiLegalPile: A 689GB multilingual legal corpus. Em *62nd Meeting of the Association for Computational Linguistics (ACL)*, 15077–15094. 10.18653/v1/2024.acl-long.805

Nunes, Rafael O., João E. Soares, Henrique D. P. dos Santos & Renata Vieira. 2019. MeSHx-notes: web-system for clinical notes. Em *1st International Workshop on Artificial Intelligence in Health (AIH)*, 5–12. 10.1007/978-3-030-12738-1_1

Nunes, Rafael O, Andre S Spritzer, Dennis G Balreira, Carla MDS Freitas & Joel L Carbonera. 2024a. An evaluation of large language models for geological named entity recognition. Em *36th International Conference on Tools with Artificial Intelligence (ICTAI)*,

Nunes, Rafael Oleques, Dennis Giovani Balreira, André Suslik Spritzer & Carla Maria Dal Sasso Freitas. 2024b. A named entity recognition approach for Portuguese legislative texts using self-learning. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 290–300.

Păiș, Vasile, Maria Mitrofan, Carol Luca Gasan, Vlad Coneschi & Alexandru Ianov. 2021. Named entity recognition in the Romanian legal domain. Em *Natural Legal Language Processing Workshop*, 9–18. 10.18653/v1/2021.nllp-1.2

Polo, Felipe Maia, Gabriel Caiaffa Floriano Mendonça, Kauê Capellato J Parreira, Lucka Gianvechio, Peterson Cordeiro, Jonathan Batista Ferreira, Leticia Maria Paz de Lima, Antônio Carlos do Amaral Maia & Renato Vicente. 2021. LegalNLP–natural language processing methods for the Brazilian legal language. arXiv [cs.CL/cs.LG]. 10.48550/arXiv.2110.15709

Reimers, Nils & Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. Em *Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3982–3992. 10.18653/v1/D19-1410

Reimers, Nils & Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4512–4525. 10.18653/v1/2020.emnlp-main.365

Rodrigues, João, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso & Tomás Osório. 2023. Advancing neural encoding of Portuguese with transformer Albertina PT-*. Em *EPIA Conference on Artificial Intelligence*, 441–453. 10.1007/978-3-031-49008-8_35

Santos, Cícero Nogueira dos & Victor Guimarães. 2015. Boosting named entity recognition with neural character embeddings. Em *5th Named Entity Workshop*, 25–33. 10.18653/v1/W15-3904

Santos, Diana, Nuno Seco, Nuno Cardoso & Rui Vilela. 2006. HAREM: An advanced NER evaluation contest for Portuguese. Em *5th International Conference on Language Resources and Evaluation (LREC)*,

Sechidis, Konstantinos, Grigorios Tsoumakas & Ioannis Vlahavas. 2011. On the stratification of multi-label data. Em *European Conference in Machine Learning and*

Knowledge Discovery in Databases, 145–158. 10.1007/978-3-642-23808-6_10

Sha, Lele, Yuheng Li, Dragan Gasevic & Guanliang Chen. 2022. Bigger data or fairer data? augmenting BERT via active sampling for educational text classification. Em *29th International Conference on Computational Linguistics (COLING)*, 1275–1285.

Shapiro, Samuel Sanford & Martin B Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika* 52(3/4). 591–611. 10.2307/2333709

Silva, Nádia, Marília Silva, Fabíola Pereira, João Tarrega, João Beinotti, Márcio Fonseca, Francisco Andrade & André Carvalho. 2021. Evaluating topic models in Portuguese political comments about bills from Brazil's chamber of deputies. Em *10th Brazilian Conference on Intelligent Systems (BRACIS)*, 104–120. 10.1007/978-3-030-91699-2_8

Souza, Fábio, Rodrigo Nogueira & Roberto Lotufo. 2020. BERTimbau: pretrained BERT models for Brazilian Portuguese. Em *9th Brazilian Conference on Intelligent Systems (BRACIS)*, 403–417. 10.1007/978-3-030-61377-8_28

Sultanum, Nicole, Devin Singh, Michael Brudno & Fanny Chevalier. 2018. Doccurate: A curation-based approach for clinical text visualization. *IEEE Transactions on Visualization and Computer Graphics* 25(1). 142–151. 10.1109/TVCG.2018.2864905

Tran, Van Cuong, Ngoc Thanh Nguyen, Hamido Fujita, Dinh Tuyen Hoang & Dosam Hwang. 2017. A combination of active learning and self-learning for named entity recognition on Twitter using conditional random fields. *Knowledge-Based Systems* 132. 179–187. 10.1016/j.knosys.2017.06.023

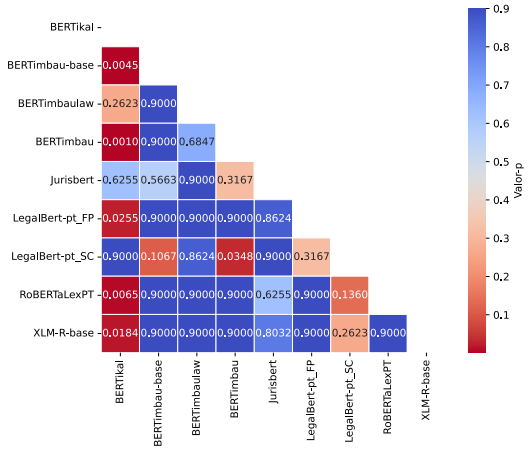
Viegas, Charles F. O., Bruno C. Costa & Renato P. Ishii. 2023. JurisBERT: a new approach that converts a classification corpus into an STS one. Em *International Conference on Computational Science and Its Applications*, 349–365. 10.1007/978-3-031-36805-9_24

Wilcoxon, Frank. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin* 1(6). 80–83. 10.2307/3001968

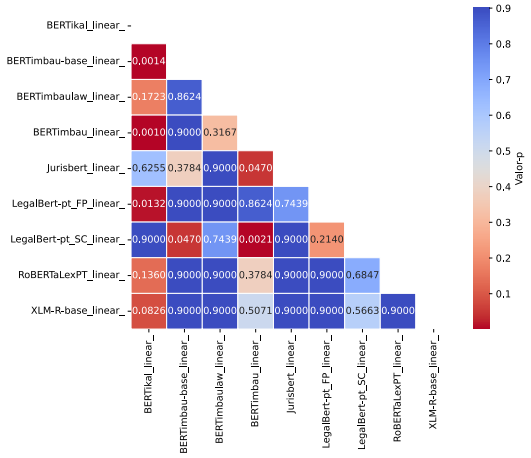
Zanuz, Luciano & Sandro José Rigo. 2022. Fostering judiciary applications with new fine-tuned models for legal named entity recognition in Portuguese. Em *15th International Conference on Computational Processing of the Portuguese Language (PROPOR)*, 219–229. 10.1007/978-3-030-98305-5_21

A. Resultados dos testes estatísticos

Neste apêndice, apresentamos os resultados completos dos testes estatísticos discutidos na Seção 6. Enquanto na seção principal os testes foram apresentados de forma resumida, com abstrações visuais, as Figuras 6a, 6b, 7a e 7b exibem diretamente os valores-p, oferecendo uma visão mais detalhada e precisa dos resultados. A diagonal principal foi omitida, pois os valores de um modelo comparado consigo mesmo são sempre iguais. Da mesma forma, os valores acima da diagonal principal foram omitidos, uma vez que são redundantes, sendo espelhados dos valores abaixo da diagonal, o que contribui para uma apresentação mais clara e objetiva dos dados.

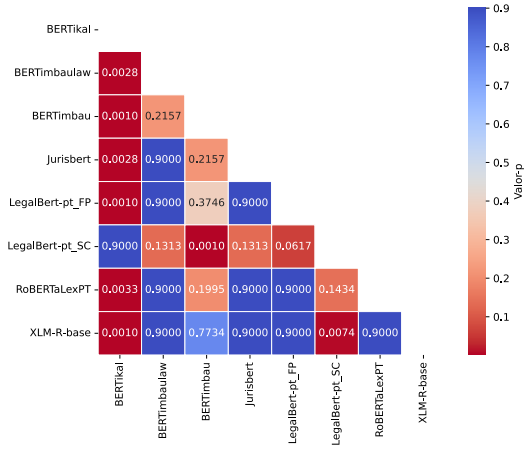


(a) Versão descontaminada.

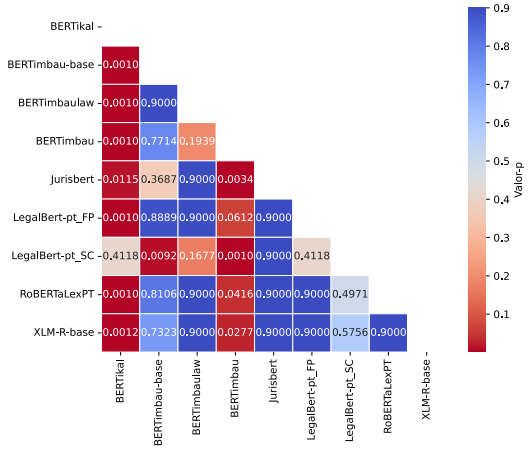


(b) Versão contaminada.

Figura 6: Mapas de calor com os resultados do teste de Friedman com o teste de Nemenyi como *post-hoc* para o ***sub-corpus* de categorias**. O modelo BERTimbau referido na figura corresponde ao modelo *large*.



(a) Versão descontaminada.



(b) Versão contaminada.

Figura 7: Mapas de calor com os resultados do teste de Friedman com o teste de Nemenyi como *post-hoc* para o ***sub-corpus* de tipos**. O modelo BERTimbau referido na figura corresponde ao modelo *large*.

Proposta recebida em Outubro 2024 e aceite para publicação em Novembro 2024.

Extração de Informação Aberta com LLM para a Língua Portuguesa

Open Information Extraction with LLM for the Portuguese Language

Bruno Cabral
Universidade Federal da Bahia

Marlo Souza
Universidade Federal da Bahia

Daniela Barreiro Claro
Universidade Federal da Bahia

Resumo

Neste estudo, investigamos a aplicação de Modelos de Linguagem de Grande Escala (LLMs) para Extração de Informação Aberta (EIA) em língua portuguesa. Enquanto a maioria dos métodos de EIA foi desenvolvida visando a língua inglesa, poucos trabalhos na literatura exploram cenários multilíngues e interlinguísticos. Embora haja um crescente interesse em métodos de EIA para o português, o uso de LLMs especificamente focados em EIA nesta língua ainda é pouco explorado. Analisamos a viabilidade de incorporar LLMs abertos e comerciais utilizando engenharia de *prompts* com poucos exemplos para EIA em português. Fornecemos uma análise detalhada do desempenho desses LLMs em tarefas de EIA, demonstrando que eles alcançam métricas de desempenho comparáveis aos sistemas de última geração. Além disso, refinamos e lançamos um LLM aberto para EIA, denominado PortOIE-Llama, que supera os LLMs comerciais em nossos experimentos. Nossos resultados destacam o potencial dos LLMs em tarefas de EIA em português e sugerem que um refinamento e ajuste fino de modelos maiores podem aprimorar ainda mais esses resultados.

Palavras chave

EIA; LLM; extração de informação; corpus

Abstract

In this study, we investigate the application of Large Language Models (LLMs) for Open Information Extraction (OpenIE) in the Portuguese language. While most OpenIE methods have been developed with a focus on the English language, few works in the literature explore multilingual and cross-linguistic scenarios. Although there is a growing interest in OpenIE methods for Portuguese, the use of LLMs specifically focused on OpenIE in this language remains underexplored. We analyze the feasibility of incorporating both open and commercial LLMs using few-shot prompt engineering for OpenIE in Portuguese. We provide a detailed analysis of the performance of these LLMs in OpenIE tasks, demonstrating that they achieve performance metrics comparable to state-of-the-art systems. Additionally, we refine and release an open LLM for OpenIE, named PortOIE-Llama, which outperforms commercial LLMs in our experiments. Our results highlight the potential of LLMs in OpenIE tasks in Portuguese and suggest that further refinement and fine-tuning of larger models can further enhance these outcomes.

Keywords

LLM; OpenIE; information extraction; corpus

1. Introdução

A Extração de Informação Aberta (EIA) é uma tarefa de Processamento de Linguagem Natural (PLN) que identifica estruturas semânticas a partir de dados não estruturados (Etzioni et al., 2008). Os sistemas de EIA extraem fatos em linguagem natural que representam relações semânticas entre entidades.

Desde 2007, com o TEXTRUNNER (Etzioni et al., 2008), vários sistemas de EIA foram propostos para diferentes idiomas. Esses sistemas tiveram diferentes tipos de abordagens, sendo a maioria baseada em abordagens não supervisionadas ou em regras criadas manualmente para identificar relações (Glauber et al., 2018). De acordo com Fader et al. (2011), os sistemas de Extração de Relações Tradicionais geram um modelo capaz de extrair cada relação alvo a partir de exemplos de treinamento anotados. Esses sistemas dependem do domínio de aplicação e sua adaptação a um novo domínio requer extenso trabalho manual. Com o objetivo de superar esses problemas, Etzioni et al. (2008) introduziu o paradigma de extração independente de domínio que emprega padrões generalizados para extrair os relacionamentos entre entidades (Li et al., 2011).

Recentemente, no entanto, impulsionados pelos avanços das redes neurais, além da disponibilidade de novos corpora, métodos de EIA baseados em aprendizado supervisionado foram propostos (Stanovsky et al., 2018; Cui et al., 2018; Sun et al., 2018; Zhang et al., 2017), alcançando novos resultados para a língua inglesa e avançando o estado da arte.

DOI: 10.21814/lm.16.2.454
This work is Licensed under a Creative Commons Attribution 4.0 License

Especificamente nos últimos anos, avanços significativos nos modelos de redes neurais, principalmente modelos gerativos, tais como o GPT-3 (Brown et al., 2020), foram impulsionados pela disponibilidade de dados e pelo poder computacional necessário para processá-los, que foram barateados (Gozalo-Brizuela & Garrido-Merchan, 2023). Progressos notáveis foram observados no Processamento de Linguagem Natural e na geração de imagens digitais. O ChatGPT, por exemplo, alcançou o título de "Aplicativo de Crescimento Mais Rápido de Todos os Tempos" ao acumular 100 milhões de usuários ativos mensais em apenas dois meses.[1]

Embora a maioria dos avanços ocorra na língua inglesa, conforme observado por Glauber et al. (2018), isso pode introduzir um viés em direção a características específicas desse idioma. Como apontado por Bender (2009, 2019), essa ênfase no inglês pode limitar a aplicabilidade dos métodos a outros idiomas. Portanto, é essencial validar os métodos em um conjunto diversificado de idiomas. No entanto, a realização de uma verificação multilíngue é desafiadora devido à escassez de corpora disponíveis para EIA em diferentes línguas. A criação manual desses corpora é complexa (Glauber et al., 2018; Léchelle et al., 2018), principalmente pela falta de uniformização dos conceitos na tarefa de EIA (Xavier et al., 2015; Léchelle et al., 2018; Stanovsky & Dagan, 2016).

Para a língua portuguesa, avanços significativos na EIA ocorreram nos últimos anos Sena & Claro (2019, 2020); Cabral et al. (2022), embora a aplicação de Modelos de Linguagem de Grande Escala (LLMs) ainda seja pouco explorada. Os LLMs demonstraram a capacidade de gerar textos mais próximos aos humanos, indicando um caminho promissor para tarefas de EIA. Este trabalho examina o potencial da utilização de LLMs comerciais e abertos para a tarefa de EIA em português.

Neste trabalho, relatamos nossa experiência na investigação do desempenho de modelos de linguagem de grande escala (LLMs) para a tarefa de EIA em português. Para isso, analisamos o desempenho dos principais modelos abertos e comerciais, e propomos um método de ajuste de LLMs para EIA em português, que consiste em ajustar um modelo de linguagem de grande escala pré-treinado para a tarefa de EIA em português.

[1]ChatGPT estabelece recorde para a base de usuários que mais cresce, https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/, acessado em 5 de novembro de 2023

Além disso, analisamos o uso de ajuste de modelos (fine-tuning) para a tarefa de EIA. Diferentes desafios foram impostos, tais como: a criação de um corpus representativo com tamanho suficiente para a tarefa de EIA em português, a criação de um *prompt* eficaz e a avaliação dos modelos ajustados.

A principal contribuição deste trabalho é a análise, avaliação do desempenho e disponibilização de um LLM ajustado para EIA (Llama-3-PortOIE) e a metodologia de criação de corpus de EIA em português de forma semiautomatizada.

Este artigo está estruturado da seguinte forma: A Seção 2 revisa os trabalhos relacionados; a Seção 3 descreve a metodologia e abordagem empregadas; a Seção 4 apresenta nossos experimentos, resultados e discussões; e a Seção 5 conclui nossos achados e discute direções futuras de pesquisa.

2. Trabalhos Relacionados

A Extração de Informação Aberta tem se tornado um campo de pesquisa cada vez mais relevante na área de Processamento de Linguagem Natural. Com o avanço e a popularização das Redes Neurais em sistemas de EIA (Stanovsky et al., 2018; Cui et al., 2018; Sun et al., 2018; Zhang et al., 2017), novos desafios e oportunidades surgiram, incluindo a necessidade de corpora anotados, mesmo para línguas amplamente estudadas como o inglês (Claro et al., 2019).

Neste contexto de evolução constante, Zhou et al. (2022) realizou uma análise abrangente dos métodos de EIA, identificando duas categorias principais de abordagens: modelos baseados em etiquetagem de sequência (Sequence tagging) e modelos gerativos. Esta categorização fornece uma estrutura útil para compreender o panorama atual da pesquisa em EIA e suas direções futuras.

Os modelos baseados em etiquetagem formulam o problema do EIA como uma tarefa de etiquetagem de sequência (*Sequence tagging*), atribuindo a cada palavra ou unidade linguística (*token*) em uma sentença uma etiqueta que indica sua função (como argumento e predicado) (Zhou et al., 2022). Nesse modelo, cada token na sentença recebe uma marcação específica que identifica seu papel na estrutura da informação extraída. As etiquetas mais comuns são "argumento" (para os elementos sobre os quais se fala) e "predicado" (para a ação ou relação entre os argumentos).

Por exemplo, o predicado

$$Caçou(o\ gato, o\ rato)$$

na sentença "O gato caçõu o rato" pode ser codificado através de etiquetas, tal como representado na Figura 1, utilizando a codificação BIO, na qual a etiqueta 'B-ARG' indica o início de um argumento, 'I-ARG' a continuação de um argumento, e 'PRED' o predicado.

O → B-ARG1
gato → I-ARG1
caçou → PRED
o → B-ARG2
rato → I-ARG2

Figura 1: Exemplo de etiquetagem de tokens em uma sentença, com a notação BIO (Begin-Inside-Outside) em que 'B' representa o início de um fragmento, 'I' indica a continuação do mesmo, e 'O' significa que a palavra não faz parte de um fragmento.

Esses modelos geralmente incorporam uma camada de *embedding* para representações de *tokens*, um *encoder* para representações contextuais de *tokens* como o BERT (Devlin et al., 2018) e um decodificador de etiquetas para a previsão de etiquetas com base na representação do *token* e no esquema de etiquetagem, como um classificador Conditional Random Fields (Lafferty et al., 2001).

Stanovsky et al. (2018) propuseram neurais para, introduzindo uma arquitetura baseada em Redes Neurais Recorrentes (RNN), considerando tal tarefa como uma etiquetagem de sequência semelhante ao Reconhecimento de Entidades Nomeadas. A partir de 2020, vários trabalhos empregaram arquiteturas Transformer diretamente ou em conjunto com o BERT (Devlin et al., 2018). Hohenecker et al. (2020) analisaram várias arquiteturas de EIA baseadas em redes neurais e introduziram um modelo de *embedding* ALBERT (Lan et al., 2020).

Por outro lado, abordagens gerativas para EIA modelam o problema como geração de sequência que produz uma sequência de extrações (Cui et al., 2018). Autores em Cui et al. (2018) e Zhang et al. (2017) exploraram essa abordagem, empregando uma estrutura de codificador-decodificador para aprender argumentos de alta confiança e tuplas de relação inicializadas a partir de um sistema EIA. Estudos contemporâneos integraram embeddings BERT em seus modelos gerativos. Por exemplo, Kolluru et al. (2020a) e Kolluru et al. (2020b) lançaram OpenIE6 e IMoJIE, respectivamente, para a língua inglesa, empregando embeddings BERT e um decodificador LSTM para abordar o problema de extrações redundantes em modelos gerativos de EIA.

Os esforços para desenvolver sistemas EIA interlinguísticos e multilíngues têm sido relativamente escassos. Os autores Zhang et al. (2017) propuseram uma abordagem interlinguística semi-supervisionada, enquanto Multi2OIE (Ro et al., 2020) utilizou M-BERT para *embedding* e extração de predicados, construindo assim extratores para múltiplas línguas, incluindo inglês, português e espanhol.

Os sistemas EIA para a língua portuguesa apresentaram diferentes abordagens ao longo do tempo. Inicialmente, foram desenvolvidos métodos baseados em regras para análise de dependência (Oliveira et al., 2022) e padrões linguisticamente orientados (Sena & Claro, 2019, 2020), que ofereciam uma ampla cobertura de fenômenos linguísticos. Mais recentemente, surgiram aplicações de aprendizado supervisionado com redes neurais profundas, como visto em trabalhos como Multi2OIE (Ro et al., 2020) e PortNOIE (Cabral et al., 2022). Essas abordagens neurais, embora focadas em tarefas mais específicas, demonstraram resultados promissores dentro de seus escopos definidos.

A aplicação de LMs para EIA é uma tendência crescente. Existem usos em diversos campos, como em Resposta Automática a Perguntas (Question Answering) e Extração de Relações e Extração de Informação. Autores em Xu et al. (2023) exploraram a aplicação de LLMs para extração de relações com poucos exemplos (*few-shots*). Oppenlaender & Hämäläinen (2023), por outro lado, investigaram a aplicação de um LLM para resposta a perguntas sobre um corpus de texto em larga escala, obtendo resultados promissores. Wei et al. (2023b) examinaram o uso de sistemas LLM para extração de informação sem exemplos (*zero-shot*), propondo enquadrá-lo como um problema de resposta a perguntas de múltiplas etapas. Kolluru et al. (2022) investigaram o uso de Modelos de Linguagem, nomeadamente BERT e mT5 Xue et al. (2021), para criar um modelo gerativo de EIA em duas etapas, que inicialmente identifica relações e depois monta as extrações para cada relação. Por fim, Cabral et al. (2024) realizaram o ajuste fino utilizando datasets existentes em um LLM em português.

Diferente das abordagens acima, o presente trabalho explora técnicas de criação de datasets de forma semiautomática, e analisa LLMs abertas e comerciais através de engenharia de *prompts* e ajuste de modelos, para avaliar sua viabilidade para EIA em português. Especificamente, analisou-se o impacto do corpus na qualidade do modelo LLM para EIA em Língua Portuguesa.

3. Construção de Corpora para Extração de Informação Aberta

A qualidade dos conjuntos de dados desempenha um papel crucial no treinamento de modelos para tarefas de PLN, especialmente em tarefas complexas como a EIA. No caso de modelos de linguagem, é necessário um volume adequado de dados para que o refinamento (fine-tuning) seja viável (Goodfellow et al., 2016).

A construção manual desses corpora, no entanto, pode ser um processo extremamente lento e oneroso, especialmente para idiomas com menos recursos disponíveis, como o português. Para contornar essas dificuldades, analisamos a criação de corpora sintéticos empregando modelos de linguagem de larga escala e técnicas avançadas de engenharia de prompts.

Aqui, detalhamos a metodologia utilizada para desenvolver um conjunto de dados sintético robusto, destinado ao treinamento de modelos LLM para a tarefa de EIA em língua portuguesa. Todos os códigos utilizados para a criação do corpus e o corpus em si estão disponíveis no GitHub.[2]

3.1. Definição de EIA

Seja $X = (x_1, x_2, \ldots, x_n)$ uma sequência ordenada de tokens x_i que compõem uma sentença. Um extrator de triplas EIA é uma função f que mapeia X para um conjunto $Y = \{y_1, y_2, \ldots, y_j\}$, onde cada elemento y_i é uma tripla $(rel_i, arg_{1_i}, arg_{2_i})$. Cada tripla y_i representa uma relação extraída da sentença X, onde rel_i é o tipo de relação, $arg1_i$ é o primeiro argumento (geralmente o sujeito), e $arg2_i$ é o segundo argumento (geralmente o objeto). Formalmente:

$$f : X \rightarrow Y$$

onde $X = (x_1, x_2, \ldots, x_n)$ e

$$Y = \{(rel_1, arg1_1, arg2_1), \ldots, (rel_j, arg1_j, arg2_j)\}$$

Assumimos que as tuplas estão sempre no formato $y = (arg_1, rel, arg_2)$, com arg_1 e arg_2 sendo sintagmas nominais criados a partir dos tokens em X, e rel representando uma relação entre arg_1 e arg_2. Por simplicidade, como é comum na área, não consideramos extrações que consistem em relações n-árias.

3.2. Corpora Existentes

Na busca por corpora em português que não fossem oriundos de traduções, foram identificadas várias fontes relevantes para a pesquisa em EIA. Os corpora descritos a seguir foram escolhidos por serem anotados diretamente em português, oferecendo uma perspectiva autêntica e diversificada dos fenômenos linguísticos presentes na língua.

O principal corpus identificado foi o *PUD 100* (Souza et al., 2024), uma versão refinada do *PUD 200* (Souza et al., 2024), ambos baseados no corpus de Dependências Universais Paralelas (PUD) em português (Nivre et al., 2020). O *PUD 100* consiste em 100 sentenças anotadas com 136 extrações, representando uma variedade de fontes de notícias e conteúdo da Wikipedia. Este conjunto é notável pela sua diversidade e complexidade linguística, características que são fundamentais para a análise e desenvolvimento de modelos de EIA. Os corpora identificados foram:

- **Pragmático**: Derivado do trabalho de Sena & Claro (2020), este corpus é composto por 400 sentenças de notícias anotadas manualmente, resultando em 485 extrações. A natureza pragmática e o contexto específico das notícias adicionam uma camada de relevância e desafio para a tarefa de extração de informação aberta.
- **Gamalho**: Utilizando textos em português, este corpus reúne extrações feitas por cinco diferentes sistemas de EIA, posteriormente validadas por especialistas. Com 103 sentenças e 346 extrações, ele utilizou recursos linguísticos discutidos nos estudos de Del Corro & Gemulla (2013) e Gamallo & Garcia (2015), oferecendo uma rica base para análises.
- **PUD 200**: Similar ao *PUD 100*, mas em uma escala maior, este conjunto inclui 200 sentneças com 337 extrações, abrangendo tanto notícias quanto conteúdo da Wikipedia, extraídas da parte portuguesa do corpus PUD (Souza et al., 2024).

[2] https://github.com/FORMAS

	# **Sentenças**	# **Extrações**
Gamalho (Gamallo & Garcia, 2015)	103	346
Pragmático (Sena & Claro, 2020)	400	485
PUD 200 (Souza et al., 2024)	200	337
PUD 100 (Souza et al., 2024)	100	136

Tabela 1: Estatísticas do Corpora

Na Tabela 1 são apresentadas as estatísticas dos corpora utilizados. Cada um oferece uma visão sobre os desafios de processar textos em português, destacando a importância de desenvolver métodos para a tarefa de EIA. A Figura 2 apresenta um exemplo deste corpus, ilustrando a complexidade das extrações e as nuances que os modelos de EIA precisam considerar.

Frase: *Eles vão atuar no sábado, 10 de junho.*

Extração 1:
Arg_0 = *"Eles"* V = *"vão atuar em"*
Arg_1 = *"o sábado"*

Extração 2:
Arg_0 = *"Eles"* V = *"vão atuar em"*
Arg_1 = *"o sábado, 10 de junho"*

Figura 2: Exemplo de frase e suas extrações no corpus

Observe na Extração 2, uma expansão da contração "no" para "em o" e a inclusão da data completa. Essa variação ilustra a complexidade do conjunto de dados e destaca a necessidade de modelos robustos e sofisticados para lidar com tais nuances. Além disso, a segunda extração sugere que o sábado mencionado é especificamente o dia 10 de junho, o que acrescenta uma camada adicional de interpretação temporal que deve ser considerada pelos modelos de EIA.

Essa complexidade evidencia a importância de desenvolver técnicas avançadas capazes de capturar e interpretar corretamente essas variações linguísticas, garantindo uma extração de informação precisa e contextualizada.

3.3. Seleção de Dados

Para a construção do corpus sintético, foram utilizados artigos da Wikipédia (Wikipedia contributors, 2024) em português. A Wikipédia é uma fonte e diversificada de informação, abrangendo uma grande quantidade de tópicos e domínios. Além disso, os artigos da Wikipédia são licenciados sob a Creative Commons Attribution-ShareAlike License[3], o que permite utilizar o conteúdo de forma livre e legal.

A Wikipédia também disponibiliza um arquivo contendo todos os artigos em uma determinada data, conhecido como *dump* da Wikipédia.[4] Esse arquivo contém o texto completo dos artigos, além de metadados adicionais, como categorias, *links* e referências. Para selecionar os artigos, foi utilizado o *dump* em português, e desenvolvemos um módulo para consultar a API da Wikipédia a fim de obter a quantidade de visitas do mês para cada artigo. Com base nesses dados, foram selecionados os 100 artigos mais visitados. Essa abordagem garantiu que os tópicos escolhidos fossem de alto interesse e relevância para o público em geral. Os artigos selecionados abrangem uma variedade de áreas, como ciência, tecnologia, cultura, história e entretenimento, proporcionando uma base diversificada para a construção do corpus sintético.

Para a criação desse corpus, os dados foram obtidos em Fevereiro de 2024, e os artigos mais visitados nesse mês. O artigo mais visitado foi "Facebook" com 1.850.173 visitas, seguido por "Jogo do bicho", "Brasil" e "Akira Toriyama", o que mostra a diversidade de tópicos abordados.

3.4. Pré-Processamento

Os artigos da Wikipédia presentes no *dump* contêm uma variedade de elementos de formatação, como macros, listas e tabelas, na linguagem chamada Wikitext[5]. Esses elementos, específicos da Wikipédia, podem interferir na tarefa de EIA, introduzindo ruído e complexidade desnecessária aos textos. Para mitigar esse problema, foi desenvolvido uma série de etapas para limpeza e processamento. Existem ferramentas para o inglês, como a `wikitextparser`[6], mas não identificamos uma ferramenta equivalente para o português.

As macros presentes no texto, como mostrado no exemplo a seguir, não podem utilizar as expansões em inglês:

```
{{PEPB|Desporto|esporte}} é toda a forma de
praticar [[Exercício físico|atividade
física]]
```

[3] https://creativecommons.org/licenses/by-sa/4.0/
[4] https://dumps.wikimedia.org/
[5] https://en.wikipedia.org/wiki/Help:Wikitext
[6] https://github.com/5j9/wikitextparser

A macro `PEPB` é específica da Wikipédia em português e expande para "**Desporto**(português europeu) ou **esporte**(português brasileiro)". Essas macros precisam ser expandidas para que os textos mantenham seu sentido completo. Para abordar esse problema, foi desenvolvido um sistema específico para expandir esses termos e limpar elementos desnecessários, como links internos.

Outra tarefa necessária para preparar os dados foi remover seções dos artigos onde o texto era de baixa qualidade para a criação desse *corpus*. Foram identificadas e removidas seções que não seriam úteis para a tarefa de EIA, como "Ver também", "Referências", "Bibliografia" e "Ligações externas". Por fim, foram removidas listas e tabelas, que podem interferir na tarefa de EIA. Este processo de limpeza foi essencial para assegurar que o texto resultante fosse adequado para a extração de informação.

3.5. Geração de Dados Sintéticos

Após a preparação inicial dos textos, foi utilizado um Modelo de Linguagem comercial, o OpenAI GPT-4 (OpenAI, 2023), para gerar as anotações necessárias para o treinamento de modelos de EIA. Para isso, foi usado um *prompt* contendo os 10 primeiros exemplos extraídos do corpus PUD-200, que serviu como guia para o modelo identificar e extrair informações semelhantes dos textos limpos. O processo é ilustrado na Figura 3. Após diversas iterações, o *prompt* utilizado pode ser visto no Prompt 1.

O GPT-4 foi então solicitado a selecionar sentenças dos artigos e a criar extrações baseadas nos exemplos mostrados. Este método de poucos exemplos (*few-shot learning*) permitiu que o modelo generalizasse a partir de um pequeno número de instâncias anotadas manualmente para produzir um grande volume de anotações sintéticas coerentes.

O corpus resultante, denominado aqui como **WikiPUD-Portuguese**, consistiu em 4.451 sentenças com um total de 4.458 extrações. Este volume representa um aumento significativo em relação aos corpora existentes, permitindo um treinamento mais robusto e eficaz dos modelos LLM de EIA. A Figura 4 apresenta um exemplo deste conjunto de dados, mostrando uma sentença e suas respectivas extrações.

4. *Pipeline* PT-EIA para LLMs

O processo desenvolvido para implementar o pipeline de EIA utilizando LLMs pode ser visto de forma resumida na Figura 3.

```
Baseado nos seguintes exemplos de
   Extração de Informação Aberta:

S: "Teoricamente,␣um␣casal␣poderia␣
   abrir␣quatro␣contas␣Tesco␣e␣
   ganhar␣3%␣em␣12,000␣-␣360."

Extração 0:
ARG0="um␣casal" V="poderia␣abrir"
   ARG1="quatro␣contas␣Tesco"
Extração 1:
ARG0="um␣casal" V="poderia␣abrir"
   ARG1="quatro␣contas␣Teoricamente"

.... O RESTANTE DOS EXEMPLOS .....

Realize extrações utilizando frases
   em Português do Brasil no texto
   abaixo.
Inclua também as frases sem
   extrações.
Mantenha o formato acima para a
   saída, como por exemplo:
S: "Frase"
Extração X:
ARG0="X" V="X" ARG1="X"

{{ artigo_wikipedia }}
```

Prompt 1: Criação do corpus.

4.1. Seleção de Modelos

Avaliamos Modelos de Linguagem de Grande Escala (LLMs) tanto de código aberto quanto comerciais. Para selecionar os modelos de melhor desempenho na data de redação deste documento (Abril de 2024), utilizamos o Chatbot Arena Leaderboard (Zheng et al., 2023). Os modelos de destaque incluíram o OpenAI GPT-4 (OpenAI, 2023), Anthropic Claude 3 Opus (Anthropic, 2023), e OpenAI GPT-3.5-turbo (Brown et al., 2020), todos sendo modelos comerciais.

O acesso a esses modelos é possível apenas através de uma API REST privada, que possui um custo elevado por chamada. Entretanto, também consideramos importante avaliar o desempenho de modelos de código aberto, que oferecem acesso irrestrito e podem ser operados localmente.

No Chatbot Arena Leaderboard, os modelos de código aberto mais performáticos e com licenças permissivas no momento são o LLaMA 3 (AI@Meta, 2024) e LLaMA 2 (Touvron et al., 2023). Esses modelos são treinados em extensos corpora de texto da internet e são capazes de prever a próxima palavra em uma sentença, o que lhes permite gerar textos com qualidade semelhante à humana a partir de entradas fornecidas.

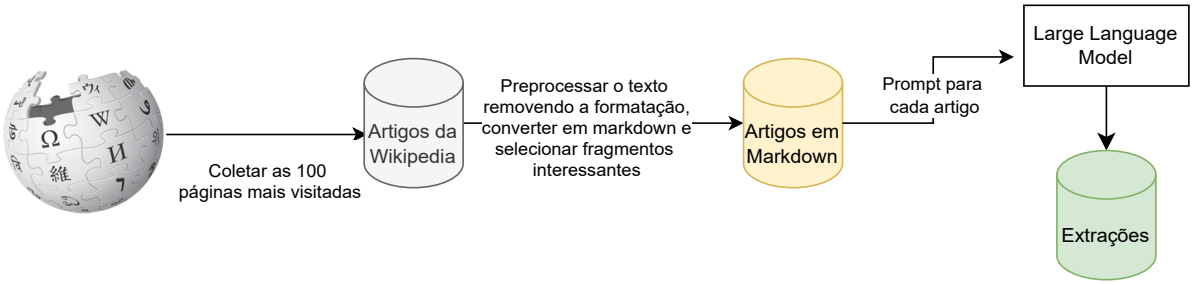


Figura 3: Diagrama ilustrando o processo de limpeza e preparação dos artigos da Wikipédia, seguido pela geração de um corpus sintético utilizando o modelo de linguagem GPT-4.

Frase: *Facebook é uma mídia social e rede social virtual lançada em 4 de fevereiro de 2004, operado e de propriedade privada da Meta, Inc.*

Extração 1:
Arg_0 = "*Facebook*" V = "*é*"
Arg_1 = "*uma mídia social e rede social virtual*"

Extração 2:
Arg_0 = "*Facebook*" V = "*foi lançado em*"
Arg_1 = "*4 de fevereiro de 2004*"

Extração 3:
Arg_0 = "*Facebook*"
V = "*é operado e de propriedade privada de*"
Arg_1 = "*Meta, Inc.*"

Figura 4: Exemplo de frase e suas extrações no corpus WikiPUD-Portuguese.

Por outro lado, os modelos comerciais geralmente alcançam desempenhos superiores devido a diversos fatores, incluindo o ajuste de alinhamento. Este processo visa tornar os modelos consistentes com as expectativas humanas. Por exemplo, o GPT-4 foi treinado utilizando um conjunto de dados baseado em instruções e também foi submetido a uma técnica de *fine-tuning* denominada *Reinforcement Learning from Human Feedback* (RLHF) para melhor alinhamento com as preferências humanas (Ouyang et al., 2022).

Os modelos mais populares à época com suporte ao Português na plataforma HuggingFace[7], como o Sabiá-7b (Pires et al., 2023) e o Lloro 7B[8] foram testados, mas os resultados não foram satisfatórios. Não conseguimos fazer com que o modelo acertasse uma única extração manualmente.

Os modelos LLaMA 3 e LLaMA 2 também possuem versões ajustadas para instruções com RLHF, conhecidas como LLaMA2 chat e LLaMA3 Instruct, respectivamente. A Tabela 2 resume os modelos utilizados neste estudo.

4.2. Ajuste Fino com Supervisão

O Ajuste Fino com Supervisão, do inglês *Supervisioned Finetuning* (SFT), é uma técnica de aprendizado de máquina que consiste em ajustar um modelo pré-treinado em uma tarefa geral para uma tarefa específica, utilizando um conjunto de dados anotado manualmente. Este método é amplamente utilizado para melhorar o desempenho de LLMs em tarefas específicas (Wei et al., 2022), permitindo que eles se adaptem melhor às nuances e requisitos particulares de um domínio ou aplicação específica.

O processo de SFT começa com um modelo que já aprendeu representações generalizáveis durante um extenso pré-treinamento em grandes conjuntos de dados, geralmente com uma variedade de textos gerais. O ajuste fino envolve a continuação do treinamento do modelo em um novo conjunto de dados que é menor, mas altamente relevante para a tarefa específica em questão. Durante o SFT, as camadas do modelo, incluindo seus pesos, são sutilmente ajustadas para minimizar o erro de previsão em relação ao conjunto de dados de destino, frequentemente envolvendo ajustes em parâmetros como a taxa de aprendizado e o número de épocas de treinamento.

O ajuste fino foi realizado utilizando o framework Axolotl.[9] Este método foi realizado em uma máquina com uma placa NVIDIA H100. O SFT oferece uma abordagem direta ao ajuste fino, permitindo atualizações mais precisas e direcionadas do modelo em comparação com métodos não supervisionados.

Utilizamos para o treinamento os *datasets* "Gamalho", "Pragmático", "PUD 200" e o *dataset* sintético descrito na seção anterior, o "WikiPUD-Portuguese". Convertemos as sentenças e extrações do *Dataset* para o formato Alpaca (Taori et al., 2023), que consiste em uma

[7] https://huggingface.co/
[8] https://huggingface.co/semantixai/Lloro
[9] Disponível em https://github.com/OpenAccess-AI-Collective/axolotl

instrução, uma entrada e uma saída. Para a instrução, utilizamos a sentença *"Dada uma sentença S, você faz extrações no formato ARG0, V, ARG1. Realize a extração para a frase abaixo:"*. Como entrada, utilizamos a sentença precedida por "S:", e a saída usa o formato: Extração N: ARG0= "..." V= "..." ARG1= "...".

Fornecemos um exemplo real de uma sentença com sua extração do conjunto de dados PUD 200:

- **instrução**: Dada uma frase S você consegue fazer extrações no formato ARG0, V, ARG1. Realize a extração para a frase abaixo:
- **entrada**: S: "Organismos que vivem em biomas marinhos devem estar adaptados ao sal presente na água."
- **saída**: Extração 0: ARG0 ="Organismos" V="vivem em" ARG1="biomas marinhos"

Para o ajuste fino, operamos o modelo Llama 3-Chat com 8B de parâmetros. As configurações principais para o treinamento incluíram o uso de um otimizador 8 bits AdamW (Loshchilov & Hutter, 2019), um tamanho de **batch** de 8, e um agendador de taxa de aprendizagem cosseno partindo de uma taxa inicial de 0,00002. Utilizamos o maior batch possível para o hardware utilizado. O treinamento também utilizou técnicas de empacotamento de amostras e **checkpointing** de gradientes, o que permitiu uma utilização eficiente da memória e aceleração do treinamento. O modelo está disponível publicamente em HuggingFace.[10]

Nome	Licença	Tamanho
OpenAI GPT-4	Comercial	N/A
Anthropic Claude 3 Opus	Comercial	N/A
OpenAI GPT-3.5	Comercial	N/A
LLaMA-2-Chat	Não comercial	7/70B
LLaMA 3	Não comercial	8/70B

Tabela 2: Resumo dos modelos LLM utilizados

4.3. Engenharia de *Prompt* para Modelos de Linguagem

A engenharia de *prompt* é essencial quando se utiliza LLMs que não passaram por um processo de ajuste fino para tarefas específicas, como a EIA. Esta técnica consiste em otimizar a formulação do *prompt* para melhorar a execução de uma tarefa específica, fornecendo instruções precisas e exemplos pertinentes (Brown et al., 2020). Essa abordagem tem se mostrado eficaz para aprimorar o desempenho dos LLMs em tarefas de PLN.

4.3.1. Metodologia para Derivação do Prompt

O desenvolvimento do *prompt* ideal para a tarefa de EIA foi um processo iterativo e sistemático. Inicialmente, foram utilizadas as primeiras cinco sentenças do conjunto de dados PUD 200 para ajustar o *prompt*. É importante destacar que essas sentenças foram utilizadas apenas como um referencial para o refinamento do *prompt*, e não como parte da avaliação ou como exemplos em uma configuração de *few-shots*.

De início, foi utilizado um *prompt* básico que instruia ao modelo a realização de extraçção de informação de uma sentenças específica. Esta abordagem inicial, revelou-se ineficaz, indicando a necessidade de instruções mais detalhadas para que o modelo compreendesse adequadamente a tarefa.

Para aprimorar essa compreensão, foi adicionado ao *prompt* um exemplo concreto de extração. Esta modificação resultou em uma melhoria significativa na capacidade do modelo de realizar a tarefa solicitada. Continuamos a refinar o *prompt*, incluindo uma definição explícita do papel do sistema, com a sentença: "Você é um sistema OpenIE extremamente inteligente e preciso...". Este ajuste demonstrou ser vantajoso, inclusive em configurações de um único exemplo (*one-shot*), sugerindo que o modelo responde bem a definições claras de seu papel.

Apesar desses avanços, observamos que a inclusão de uma definição extensiva da tarefa EIA, como a encontrada na Wikipedia, dentro do *prompt*, não melhorou os resultados para as sentenças de validação. Por fim, adotamos uma codificação dos exemplos em um formato de chave-valor com quebras de linha, o que tornou as respostas do modelo menos conversacionais, como "Sim, eu posso fazer uma extração...", e mais estruturadas, facilitando a análise subsequente.

Após várias iterações de ajustes, o *prompt* finalizado foi o seguinte, como mostrado na Figura 5.

Neste *prompt*, [**FRASE**] representa a sentença a ser processada, e [**Exemplos do PUD 200**] são alguns exemplos de extrações de EIA no contexto da sentença. Esta abordagem é conhecida como aprendizado com poucos exemplos (*few-shot learning*), no qual o modelo de linguagem é fornecido com um pequeno número de exemplos de extrações de EIA para facilitar seu entendimento da tarefa (Wang et al., 2020).

[10] https://huggingface.co/bratao/llama7b-finetuned-openie-lora

Você é um sistema muito inteligente e preciso de OpenIE. Dada uma frase S, você consegue realizar extrações no formato ARG0, V, ARG1, como por exemplo:
S: "Maria é Professora de Banco de Dados"
Extração 1:
ARG0 = 'Maria'
V = 'é'
ARG1 = 'Professora de Banco de Dados'
[**Exemplos do PUD 200**]
Realize a extração para a frase abaixo:
S: [**FRASE**]

Figura 5: Prompt final utilizado para extrações de EIA

Também exploramos outros *prompts* e técnicas, como o *prompt* de *Chain of Thought* (Cadeia de Pensamento)(Wei et al., 2023a). No entanto, em nossos experimentos, descobrimos que o *prompt* proposto consistentemente produzia saídas esperadas para os exemplos analisados. Como um trabalho futuro, fica a possibilidade de como explorar outras formas de engenharia de *prompt* em LLMs para o problema de Extração de Informação Aberta.

5. Experimentos

Nesta seção, detalhamos a validação empírica para a extração de triplas utilizando LLMs selecionados, incluindo o novo modelo ajustado, o **LLaMA-3-8B-FT**.

5.1. Configuração do ambiente

Cada etapa do *pipeline* foi implementada em Python, versão 3.10, utilizando as bibliotecas da OpenAI e Anthropic para acessar a API dos LLMs. Para LLMs locais de código aberto, foi utilizado o projeto Llama.cpp (Gerganov, 2023) para carregar os modelos e processar as saídas.

Os experimentos utilizaram como configuração para geração a *temperatura* de 0.2 e *max˙tokens* em 1000. Já os parâmetros *top˙p*, *frequency˙penalty* e *presence˙penalty* foram configurados como 0, assegurando que nenhuma penalidade fosse aplicada a tokens frequentes nas saídas.

Os modelos locais foram executados em um servidor na nuvem equipado com um processador AMD EPYC 7003, 30 vCPUs, uma GPU NVIDIA H100 com 92 GB de VRAM e 200 GB de RAM.

Cada sentença no *Corpus PUD 100* foi tokenizada e submetida a cada LLM junto com o *prompt* especificado na Seção 4.3. Os modelos geraram saídas de texto, das quais foram extraídas as triplas. O **LLaMA-3-7B-FT**, o modelo ajustado, utilizou um *prompt* personalizado idêntico ao usado em seu treinamento; portanto, não foi aplicada a estratégia de *prompt* de poucos exemplos para este modelo.

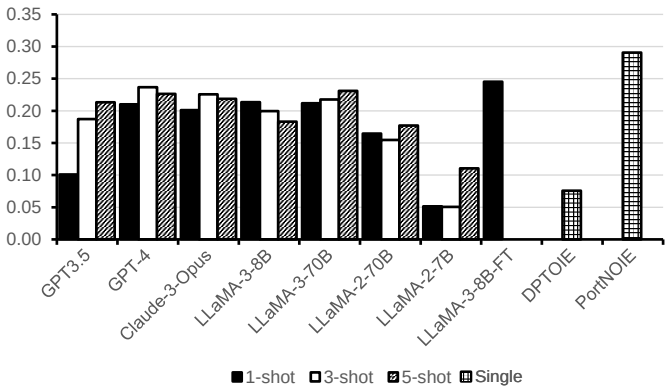
Para a comparação dos resultados com outros sistemas de EIA em português, foram selecionados o DptOIE (Oliveira et al., 2022) e o PortNOIE (Cabral et al., 2022). O DptOIE emprega uma busca em profundidade na árvore de dependência para a extração, enquanto o PortNOIE utiliza uma rede neural profunda e reporta os melhores resultados de F1 para o português.

Foram utilizadas as métricas de precisão (P), recall (R) e F1 para avaliar a qualidade dos extratores. O código de avaliação foi adaptado de Stanovsky et al. (2018), uma metodologia amplamente utilizada em estudos subsequentes (Ro et al., 2020; Kolluru et al., 2020a). Por padrão, esse benchmark emprega um método de pontuação chamado **Correspondência Léxica**, que considera as palavras da tripla como correspondentes se forem pelo menos 50% similares, independentemente da ordem. Também foi utilizada a estratégia de **Correspondência Perfeita**, que considera as strings idênticas após a remoção de pontuação.

Essas métricas comparam as triplas extraídas por cada modelo com as triplas padrão ouro no *Corpus PUD 100*. Uma correspondência exata com a tripla padrão ouro foi considerada uma correspondência. Para a correspondência léxica, foi adotada uma estratégia de correspondência mais relaxada, considerando uma correspondência se pelo menos dois componentes da tripla (arg1, rel, arg2) corresponderem ao padrão ouro.

5.2. Resultados

A análise dos resultados está organizada em duas partes. Primeiramente, são apresentados os escores F1 para correspondências perfeitas e léxica em diferentes modelos, utilizando estratégias de prompt de 1-shot, 3-shot e 5-shot. Em seguida, é realizada uma análise detalhada do desempenho dos modelos empregando sua melhor estratégia de prompt no *Corpus PUD 100*. Os resultados são exibidos na Tabela 3 e na Tabela 4, respectivamente, com uma comparação visual dos escores F1 dos diferentes modelos utilizando a melhor estratégia de prompt.



	Correspondência Perfeita F1 ↑			Correspondência Léxica F1 ↑		
Modelos	**0/1-shot**	**3-shot**	**5-shot**	**0/1-shot**	**3-shot**	**5-shot**
GPT3.5	0.0301	0.1007	0.0955	0.1005	0.1870	0.2132
GPT-4	0.1013	0.1065	0.0978	0.2094	0.2366	0.2262
Claude3-Opus	0.1003	0.1290	0.1125	0.2007	0.2258	0.2185
LLaMA-3-8B	0.0828	0.0828	0.0944	0.2130	0.1994	0.1833
LLaMA-3-70B	0.1114	0.1146	0.1375	0.2111	0.2177	0.2310
LLaMA-2-70B	0.0447	0.0619	0.0655	0.1641	0.1547	0.1770
LLaMA-2-7B	0.0255	0.0144	0.0158	0.0510	0.0505	0.1106
LLaMA-3-8B-FT (PortOIE-Llama3) (0-shot)	**0.1290**	N/A	N/A	**0.2446**	N/A	N/A
DPTOIE (0-shot)	0.0112	N/A	N/A	0.0757	N/A	N/A
PortNOIE (0-shot)	0.0979	N/A	N/A	0.2905	N/A	N/A

Tabela 3: Medidas F1 de Diferentes Modelos para o conjunto de dados PUD100 usando 1, 3 e 5 exemplos para Correspondência Perfeita e Correspondência Léxica.

Model	**Precisão ↑**	**Recall ↑**	**F1 ↑**	**Custo 1k ↓**	**Tempo ↓**
GPT3.5(5-shot)	0.2132	0.2132	0.2132	$1.20	2.7 segs
GPT-4(3-shot)	0.1980	0.2941	0.2366	$36.80	4.2 segs
Claude3-Opus(3-shot)	0.2011	0.2573	0.2258	$20.25	5.2 segs
LLaMA-3-8B (1-shot)	0.1782	0.2647	0.2130	$0.42	1.4 segs
LLaMA-3-70B (3-shot)	0.2269	0.2352	0.2310	$1.11	3.7 segs
LLaMA-2-70B(5-shot)	0.1597	0.1985	0.1770	$1.16	3.8 segs
LLaMA-2-7B(5-shot)	0.1196	0.1029	0.1106	$0.45	1.5 segs
LLaMA-3-8B-FT(PortOIE-Llama3)	0.28	0.2058	0.2446	$0.42	1.4 segs
DPTOIE	0.0408	0.0787	0.0757	$1.62	5.3 segs
PortNOIE	**0.3269**	0.2615	**0.2905**	**$0.15**	**0.5** segs

Tabela 4: Medidas F1 de Diferentes Modelos para o conjunto de dados PUD100 usando a estratégia de solicitação de melhor desempenho para Correspondência Léxica.

Considerando apenas os LLMs no cenário **Correspondência Perfeita**, o modelo LLaMA-2-7B-FT, a versão refinada do modelo LLaMA-2-7B, chamado PortOIE-Llama3, supera outros modelos em todos os cenários com uma pontuação de 0.1271 conforme Tabela 3. O modelo original, o LLaMA-2-7B, apresenta um desempenho consideravelmente pior, com o maior F1 de 0.0255, um aumento de desempenho de 5 vezes. OEste modelo refinado é melhor que o segundo melhor modelo, o modelo comercial GPT-4, com pontuações de 0.1013, 0.1065 e 0.0978.

Considerando o cenário de **Correspondência Perfeita**, o modelo LLaMA-3-8B-FT, nossa versão refinada do modelo LLaMA-3-8B, denominada PortOIE-Llama3, supera outros modelos com uma pontuação F1 de 0.1290 no cenário de 0/1-shot, conforme mostrado na Tabela 3. Este desempenho é notavelmente superior ao do modelo base LLaMA-3-8B, que alcança um F1 máximo de 0.0944 no cenário de 5-shot, demonstrando a eficácia do nosso processo de *fine-tuning*.

Entre os modelos comerciais, o Claude3-Opus apresenta o melhor desempenho em correspondência perfeita, com um F1 de 0.1290 no cenário de 3-shot, superando ligeiramente o GPT-4 (F1 de 0.1065 no 3-shot). O GPT3.5 mostra uma melhoria significativa do cenário 1-shot para o 3-shot, mas seu desempenho é inferior aos modelos mais avançados.

Na **Correspondência Léxica**, o PortOIE-Llama3 mantém sua superioridade com um F1 de 0.2446 no cenário de 0-shot. O LLaMA-3-70B apresenta um desempenho consistente, alcançando seu melhor F1 de 0.2310 no cenário de 5-shot. Entre os modelos comerciais, o GPT-4 lidera com um F1 de 0.2366 no cenário de 3-shot, seguido de perto pelo Claude3-Opus.

É interessante observar que o aumento no número de exemplos nos *prompts* nem sempre resulta em melhor desempenho. Por exemplo, o GPT-4 atinge seu pico no cenário de 3-shot e apresenta uma leve queda no 5-shot, tanto para correspondência perfeita quanto léxica. Isso sugere que a qualidade e relevância dos exemplos podem ser mais importantes do que a quantidade, alinhando-se com achados de outros estudos na literatura, como Fu et al. (2023).

Na análise de desempenho detalhada usando a melhor estratégia de prompt de cada modelo no conjunto de dados PUD100 para Correspondência Léxica (Tabela 4), observamos resultados interessantes:

- O modelo PortNOIE exibe a maior pontuação de precisão (0.3269) e a maior pontuação F1 (0.2905). Além disso, possui o menor custo por 1k tokens ($0.15) e o menor tempo médio de previsão (0.5 segundos), tornando-o o modelo mais eficiente em termos de custo-benefício.
- O PortOIE-Llama3 (LLaMA-3-8B-FT) apresenta o segundo maior F1 (0.2372) entre os LLMs, com um custo significativamente menor ($0.42 por 1k *tokens*) e um tempo de previsão mais curto (1.4 segundos) em comparação com modelos comerciais como GPT-4 e Claude3-Opus.
- O GPT-4 alcança o terceiro maior F1 (0.2366), mas com o custo mais alto ($36.80 por 1k tokens) e um tempo de previsão relativamente longo (4.2 segundos).
- O LLaMA-3-70B mostra um bom equilíbrio entre desempenho (F1 de 0.2310) e eficiência, com um custo moderado ($1.11 por 1k *tokens*) e tempo de previsão razoável (3.7 segundos).

Uma análise dos resultados revela um cenário desafiador para a tarefa de Extração de Informação Aberta em português. Os escores F1 obtidos, mesmo pelos melhores modelos, são objetivamente baixos, com o melhor resultado atingindo apenas 0,2905 (PortNOIE) para correspondência léxica. Esses números indicam que a EIA em português ainda está longe de ser um problema resolvido, possivelmente devido à complexidade inerente da língua portuguesa, com suas nuances sintáticas e semânticas, além das limitações dos modelos atuais em capturar relações semânticas complexas.

Nossos resultados destacam que modelos refinados, como o PortOIE-Llama3, podem superar modelos comerciais de ponta em tarefas específicas, oferecendo uma alternativa eficiente e de baixo custo. Um aspecto particularmente notável é o desempenho superior do PortNOIE, um sistema especializado, sobre os LLMs genéricos. Isto sugere que, no atual estado da tecnologia, abordagens específicas para domínio ainda podem ser mais efetivas que modelos de linguagem grandes e genéricos para tarefas específicas como EIA.

A variação no desempenho entre diferentes estratégias de *prompt* (1-shot, 3-shot, 5-shot) indica que a otimização de *prompts* é crucial e que mais exemplos nem sempre levam a melhores resultados. Isso ressalta a importância de uma seleção cuidadosa e experimentação com diferentes estratégias de *prompt* para cada modelo e tarefa específica.

Por fim, ao considerar a implementação prática desses modelos, é essencial equilibrar o desempenho com fatores como custo e velocidade de inferência. Modelos como PortNOIE e PortOIE-Llama3 oferecem um excelente equilíbrio entre esses fatores, tornando-os opções atraentes para aplicações em larga escala ou com restrições de recursos.

5.3. Análise Qualitativa

Os resultados qualitativos foram avaliados para os modelos de linguagem na tarefa de Extração de Informação Aberta. A sentença de entrada foi: "Teoricamente, um casal poderia abrir quatro contas Tesco e ganhar 3% em £12,000—£360", e as extrações esperadas eram de acordo com o corpus PUD 100 as da Figura 6.

GPT-3.5: Este sistema respondeu que não conseguiu realizar a tarefa, pois não é projetado para Extração de Informação Aberta. Sugeriu o uso de outras ferramentas ou bibliotecas para esse propósito.

Extração 0:
Arg_0 = "*um casal*" V = "*poderia abrir*"
Arg_1 = "*quatro contas Tesco*"
Extração 1:
Arg_0 = "*um casal*" V = "*poderia abrir*"
Arg_1 = "*quatro contas Teoricamente*"

Figura 6: Extrações esperadas de acordo com o corpus PUD 100.

GPT-4: Este sistema teve um bom desempenho, mas extraiu informações adicionais que não faziam parte da saída esperada. A saída foi a apresentada na Figura 7.

Arg_0 = "*um casal*"
V = "*poderia abrir*"
Arg_1 = "*quatro contas Tesco*"
Arg_2 = "*e ganhar*"
Arg_3 = "*3% em £12,000 – £360*"

Figura 7: Extração realizada pelo GPT-4.

Embora tenha identificado corretamente o Arg_0, V e Arg_1 da primeira extração esperada, adicionou informações extras como Arg_2 e Arg_3.

Claude-3-Opus: Este sistema demonstrou uma compreensão mais sofisticada, fornecendo duas extrações relevantes (Figura 8).

Extração 1:
Arg_0 = "*um casal*"
V = "*poderia abrir*"
Arg_1 = "*quatro contas Tesco*"

Extração 2:
Arg_0 = "*um casal*"
V = "*ganhar*"
Arg_1 = "*3% em £12,000 – £360*"

Figura 8: Extrações realizadas pelo Claude-3-Opus.

A primeira extração corresponde exatamente à extração 0 esperada, enquanto a segunda captura informações adicionais relevantes da sentença.

LLaMA-3-8B: Este modelo forneceu múltiplas extrações, demonstrando uma abordagem mais abrangente (Figura 9).

Embora tenha capturado aspectos relevantes, nenhuma das extrações corresponde exatamente às esperadas. A extração 3, em particular, apresenta uma interpretação incorreta.

Extração 1:
Arg_0 = "*um casal*"
V = "*poderia*"
Arg_1 = "*abrir quatro contas Tesco e ganhar 3% em £12,000 – £360*"

Extração 2:
Arg_0 = "*um casal*"
V = "*ganhar*"
Arg_1 = "*3% em £12,000 – £360*"

Extração 3:
Arg_0 = "*Tesco*"
V = "*abrir*"
Arg_1 = "*quatro contas*"

Figura 9: Extrações realizadas pelo LLaMA-3-8B.

LLaMA-3-70B: Este modelo forneceu duas extrações precisas de acordo com a Figura 10.

Extração 1:
Arg_0 = "*um casal*"
V = "*poderia abrir*"
Arg_1 = "*quatro contas Tesco*"

Extração 2:
Arg_0 = "*um casal*"
V = "*ganhar*"
Arg_1 = "*3% em £12,000 – £360*"

Figura 10: Extrações realizadas pelo LLaMA-3-70B.

A primeira extração corresponde exatamente à extração 0 esperada, enquanto a segunda captura colocou como V = "*ganhar*", omitindo o "*poderia*".

LLaMA-2-70B: Este modelo produziu duas extrações, ambas relevantes, mas com algumas diferenças em relação ao esperado (Figura 11)

Extração 1:
Arg_0 = "*um casal*"
V = "*poderia*"
Arg_1 = "*abrir quatro contas Tesco*"

Extração 2:
Arg_0 = "*ganhar*"
V = "*3%*"
Arg_1 = "*£12,000 – £360*"

Figura 11: Extrações realizadas pelo LLaMA-2-70B.

A primeira extração é próxima da esperada, mas separa "*poderia*" e "*abrir.*" A segunda extração apresenta uma interpretação diferente da estrutura da sentença.

LLaMA-2-7B: Este modelo não conseguiu realizar a tarefa, demonstrando uma compreensão limitada do problema e solicitando mais entrada em vez de fornecer uma extração.

LLaMA-3-8B-FT (PortOIE-Llama3): Este modelo ajustado com o corpus proposto para a tarefa, teve um bom desempenho, produzindo extrações muito próximas das esperadas (Figura 12).

Extração 0:
Arg_0 = "*um casal*" V = "*poderia abrir*"
Arg_1 = "*quatro contas Tesco*"
Extração 1:
Arg_0 = "*um casal*" V = "*ganhar*"
Arg_1 = "*3% em £12,000 – £360*"

Figura 12: Extrações realizadas pelo LLaMA-3-8B-FT (PortOIE-Llama3).

A extração 0 corresponde exatamente à esperada, enquanto a extração 1 cometeu o mesmo erro do LLaMA-3-70B, omitindo o "*poderia*".

Em resumo, LLaMA-3-8B-FT, LLaMA-3-70B e Claude-3-Opus demonstraram os melhores desempenhos, produzindo extrações precisas e relevantes. GPT-4 e LLaMA-2-70B também tiveram bom desempenho, mas com algumas inconsistências. LLaMA-3-8B forneceu extrações variadas, mas nem todas precisas. GPT-3.5 e LLaMA-2-7B não conseguiram realizar a tarefa adequadamente. Notavelmente, o ajuste do LLaMA-3-8B com o corpus proposto resultou em uma melhoria significativa no desempenho, fazendo-o superar modelos muito maiores e destacando a importância da adaptação específica para a tarefa.

5.4. Limitações

A primeira limitação de nossa abordagem é o conjunto de dados relativamente compacto, que pode limitar a ampla aplicabilidade de nossas conclusões. Além disso, a qualidade do *prompt* influencia diretamente a eficácia dos LLMs. Como mencionado anteriormente, *prompts* variados podem produzir resultados diversos. Quanto à tarefa, a estrutura de extração de relações binárias que empregamos pode não capturar completamente a complexidade de algumas sentenças. Além disso, o conjunto de dados empregado se baseia apenas em relações binárias. Por último, os LLMs podem estar sujeitos a vieses intrínsecos nos dados de treinamento, potencialmente afetando a qualidade e a justiça das tarefas de EIA.

6. Conclusão e Trabalhos Futuros

Esta pesquisa explorou a eficácia dos Modelos de Linguagem de Grande Escala (LLMs) no contexto da Extração de Informação Aberta (EIA) para o Português. Foram realizados experimentos empregando diversas estratégias de *prompt* e comparando o desempenho de vários modelos, nomeadamente GPT-4, GPT-3.5, LLaMA-3, Claude-3 Opus, LLaMA-2, e o modelo ajustado (*fine-tuned*) proposto neste trabalho, LLaMA-3-8B-FT (PortOIE-Llama3), comparado com outros sistemas de EIA em Português como DptOIE e PortNOIE.

Os resultados revelaram que o LLM ajustado proposto (PortOIE-Llama3) superou consistentemente outros LLMs em pontuações F1 em cenários de correspondência perfeita e léxica, superando inclusive modelos comerciais de ponta como GPT-4 e Claude-3 Opus.

No entanto, apesar das altas pontuações F1 alcançadas pelos LLMs, eles permanecem intensivos em recursos. O PortNOIE demonstrou desempenho superior não apenas em termos de métricas de desempenho, mas também em custo-efetividade e velocidade de previsões, alcançando a maior pontuação de precisão, a maior pontuação F1, o menor custo para 1k previsões e o menor tempo médio de previsão. Isso sugere que, embora LLMs como GPT-4 e LLaMA-3 possam oferecer desempenho notável, um modelo especializado permanece a escolha ótima para EIA em Português.

O modelo LLaMA-3-8B, ajustado para EIA (nosso PortOIE-Llama3), demonstra um potencial significativo para exploração futura. É importante notar que este modelo foi refinado utilizando um conjunto de dados com um número limitado de exemplos em língua portuguesa, partindo do modelo original LLaMA-3, que foi treinado predominantemente com dados em língua inglesa. Considerando essas limitações, é razoável antecipar que um ajuste mais abrangente, empregando um volume maior de dados em português, juntamente com a utilização de um LLM de maior capacidade e melhor adaptado ao idioma português, além de ser especificamente projetado para a tarefa de EIA, poderia resultar em um desempenho superior.

Em conclusão, este trabalho contribui para o entendimento da aplicação de Modelos de Linguagem de Grande Escala para EIA para o Português. Os achados desta pesquisa têm implicações práticas para a criação de sistemas de EIA eficientes em relação ao custo para o Português. Pesquisas futuras poderiam explorar a

otimização usando várias estratégias de *prompt* e avaliar o desempenho desses modelos em conjuntos de dados maiores e mais diversos. Além disso, investigar o impacto de diferentes técnicas de ajuste fino e a incorporação de conhecimento específico do domínio nos LLMs para melhorar ainda mais o desempenho em tarefas de EIA seria uma direção promissora.

Este modelo está disponível publicamente no HuggingFace[11]. Os dados e o código estão disponíveis no GitHub[12].

Referências

AI@Meta. 2024. Llama 3 model card. Github.

Anthropic. 2023. Meet claude. Accessed: 2023-04-03.

Bender, Emily. 2019. English isn't generic for language, despite what NLP papers might lead you to believe [slides]. Em *Symposium on Data Science and Statistics (SDSS)*,

Bender, Emily M. 2009. Linguistically naïve != language independent: Why NLP needs linguistic typology. Em *Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, 26–32.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever & Dario Amodei. 2020. Language models are few-shot learners. arXiv [cs.CL]. 10.48550/arXiv.2005.14165

Cabral, Bruno, Daniela Claro & Marlo Souza. 2024. Exploring open information extraction for Portuguese using large language models. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 127–136.

Cabral, Bruno, Marlo Souza & Daniela Barreiro Claro. 2022. PortNOIE: A neural framework for open information extraction for the Portuguese language. Em *International Conference on Computational Processing of the Portuguese Language (PROPOR)*, 243–255. 10.1007/978-3-030-98305-5_23

Claro, Daniela, Marlo Souza, Clarissa Castellã Xavier & Leandro Oliveira. 2019. Multilingual open information extraction: Challenges and opportunities. *Information* 10(7). 228. 10.3390/info10070228

Cui, Lei, Furu Wei & Ming Zhou. 2018. Neural open information extraction. arXiv [cs.CL]. 10.48550/arXiv.1805.04270

Del Corro, Luciano & Rainer Gemulla. 2013. ClausIE: clause-based open information extraction. Em *22nd International conference on World Wide Web (WWW)*, 355–366. 10.1145/2488388.2488420

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv [cs.CL]. 10.48550/arXiv.1810.04805

Etzioni, Oren, Michele Banko, Stephen Soderland & Daniel S. Weld. 2008. Open information extraction from the web. *Communications of the ACM* 51(12). 68–74. 10.1145/1409360.1409378

Fader, Anthony, Stephen Soderland & Oren Etzioni. 2011. Identifying relations for open information extraction. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1535–1545.

Fu, Yao, Hao Peng, Ashish Sabharwal, Peter Clark & Tushar Khot. 2023. Complexity-based prompting for multi-step reasoning. arXiv [cs.CL/cs.AI/cs.LG]. 10.48550/arXiv.2210.00720

Gamallo, Pablo & Marcos Garcia. 2015. Multilingual open information extraction. Em *Portuguese Conference on Artificial Intelligence (EPIA)*, 711–722. 10.1007/978-3-319-23485-4_72

Gerganov, Georgi. 2023. LLaMA C++. GitHub repository.

Glauber, Rafael, Leandro Souza de Oliveira, Cleiton Fernando Lima Sena, Daniela Barreiro Claro & Marlo Souza. 2018. Challenges of an annotation task for open information extraction in Portuguese. Em *International Conference on Computational Processing of the Portuguese Language (PROPOR)*, 66–76. 10.1007/978-3-319-99722-3_7

[11] https://huggingface.co/bratao/Llama-PortOIE3
[12] https://github.com/FORMAS/openie_generative

Goodfellow, Ian J., Yoshua Bengio & Aaron Courville. 2016. *Deep learning*. MIT Press

Gozalo-Brizuela, Roberto & Eduardo C. Garrido-Merchan. 2023. ChatGPT is not all you need. a state of the art review of large generative AI models. arXiv [cs.LG/cs.AI]. 10.48550/arXiv.2301.04655

Hohenecker, Patrick, Frank Mtumbuka, Vid Kocijan & Thomas Lukasiewicz. 2020. Systematic comparison of neural architectures and training approaches for open information extraction. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 8554–8565. 10.18653/v1/2020.emnlp-main.690

Kolluru, Keshav, Vaibhav Adlakha, Samarth Aggarwal, Soumen Chakrabarti et al. 2020a. OpenIE6: Iterative grid labeling and coordination analysis for open information extraction. arXiv [cs.CL]. 10.48550/arXiv.2010.03147

Kolluru, Keshav, Samarth Aggarwal, Vipul Rathore, Mausam & Soumen Chakrabarti. 2020b. IMoJIE: Iterative memory-based joint open information extraction. Em *58th Meeting of the Association for Computational Linguistics (ACL)*, 5871–5886. 10.18653/v1/2020.acl-main.521

Kolluru, Keshav, Muqeeth Mohammed, Shubham Mittal, Soumen Chakrabarti & Mausam . 2022. Alignment-augmented consistent translation for multilingual open information extraction. Em *60th Meeting of the Association for Computational Linguistics (ACL)*, 2502–2517. 10.18653/v1/2022.acl-long.179

Lafferty, John D., Andrew McCallum & Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. Em *International Conference on Machine Learning*, 282–289.

Lan, Zhenzhong, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma & Radu Soricut. 2020. ALBERT: A Lite BERT for self-supervised learning of language representations. Em *International Conference on Learning Representations*,

Léchelle, William, Fabrizio Gotti & Philippe Langlais. 2018. WiRe57: A fine-grained benchmark for open information extraction. arXiv [cs.CL]. 10.48550/arXiv.1809.08962

Li, Haibo, Danushka Bollegala, Yutaka Matsuo & Mitsuru Ishizuka. 2011. Using graph based method to improve bootstrapping relation extraction. Em *Computational Linguistics and Intelligent Text Processing (CICLing)*, 127–138. 10.1007/978-3-642-19437-5_10

Loshchilov, Ilya & Frank Hutter. 2019. Decoupled weight decay regularization. arXiv [cs.LG]. 10.48550/arXiv.1711.05101

Nivre, Joakim, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers & Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. Em *12th Language Resources and Evaluation Conference (LREC)*, 4034–4043.

Oliveira, Leandro, Daniela Barreiro Claro & Marlo Souza. 2022. DptOIE: A portuguese open information extraction based on dependency analysis. *Artificial Intelligence Review* 56(7). 7015–7046. 10.1007/s10462-022-10349-4

OpenAI. 2023. GPT-4 technical report. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2303.08774

Oppenlaender, Jonas & Joonas Hämäläinen. 2023. Mapping the challenges of HCI: An application and evaluation of ChatGPT and GPT-4 for cost-efficient question answering. arXiv [cs.HC/cs.AI]. 10.48550/arXiv.2306.05036

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike & Ryan Lowe. 2022. Training language models to follow instructions with human feedback. arXiv [cs.CL/cs.AI/cs.LG]. 10.48550/arXiv.2203.02155

Pires, Ramon, Hugo Abonizio, Thales Sales Almeida & Rodrigo Nogueira. 2023. Sabiá: Portuguese large language models. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 226–240. 10.1007/978-3-031-45392-2_15

Ro, Youngbin, Yukyung Lee & Pilsung Kang. 2020. Multi^{2}OIE: Multilingual open information extraction based on multi-head attention with BERT. arXiv [cs.CL/cs.LG]. 10.48550/arXiv.2009.08128

Sena, Cleiton Fernando Lima & Daniela Barreiro Claro. 2019. InferPortOIE: A Portuguese open information extraction system with inferences. *Natural Language Engineering* 25(2). 287–306. 10.1017/S135132491800044X

Sena, Cleiton Fernando Lima & Daniela Barreiro Claro. 2020. PragmaticOIE: A pragmatic open information extraction for Portuguese language. *Knowledge and Information Systems* 62. 3811–3836. 10.1007/s10115-020-01442-7

Souza, Marlo, Bruno Cabral, Daniela Claro & Lais Salvador. 2024. Challenges in expanding Portuguese resources: A view from open information extraction. arXiv [cs.CL]. 10.48550/arXiv.2501.11851

Stanovsky, Gabriel & Ido Dagan. 2016. Creating a large benchmark for open information extraction. Em *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2300–2305. 10.18653/v1/D16-1252

Stanovsky, Gabriel, Julian Michael, Luke Zettlemoyer & Ido Dagan. 2018. Supervised open information extraction. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 885–895. 10.18653/v1/N18-1081

Sun, Mingming, Xu Li, Xin Wang, Miao Fan, Yue Feng & Ping Li. 2018. Logician: a unified end-to-end neural approach for open-domain information extraction. Em *11th ACM International Conference on Web Search and Data Mining*, 556–564. 10.1145/3159652.3159712

Taori, Rohan, Ishaan Gulrajani, Tianhao Zhang, Yves Dubois, Xiang Li, Carlos Guestrin, Percy Liang & Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. Relatório técnico. Stanford Center for Research on Foundation Models.

Touvron, Hugo, Louis Martin & the Llama 2 Team. 2023. Llama 2: Open foundation and fine-tuned chat models. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2307.09288

Wang, Yaqing, Quanming Yao, James Kwok & Lionel M. Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. arXiv [cs.LG/cs.AI]. 10.48550/arXiv.1904.05046

Wei, Jason, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai & Quoc V. Le. 2022. Finetuned language models are zero-shot learners. arXiv [cs.CL]. 10.48550/arXiv.2109.01652

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le & Denny Zhou. 2023a. Chain-of-thought prompting elicits reasoning in large language models. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2201.11903

Wei, Xiang, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang & Wenjuan Han. 2023b. Zero-shot information extraction via chatting with ChatGPT. arXiv [cs.CL]. 10.48550/arXiv.2302.10205

Wikipedia contributors. 2024. Wikipedia, the free encyclopedia. [Online; accessed 9-June-2024].

Xavier, Clarissa Castellã, Vera Lúcia Strube de Lima & Marlo Souza. 2015. Open information extraction based on lexical semantics. *Journal of the Brazilian Computer Society* 21. 4. 10.1186/s13173-015-0023-2

Xu, Xin, Yuqi Zhu, Xiaohan Wang & Ningyu Zhang. 2023. How to unleash the power of large language models for few-shot relation extraction? arXiv [cs.CL/cs.AI/cs.DB/cs.IR/cs.LG]. 10.48550/arXiv.2305.01555

Xue, Linting, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua & Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. arXiv [cs.CL]. 10.48550/arXiv.2010.11934

Zhang, Sheng, Kevin Duh & Benjamin Van Durme. 2017. MT/IE: Cross-lingual open information extraction with neural sequence-to-sequence models. Em *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 64–70.

Zheng, Lianmin, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez & Ion Stoica. 2023. Judging LLM-as-a-judge with MT-Bench and chatbot Arena. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2306.05685

Zhou, Shaowen, Bowen Yu, Aixin Sun, Cheng Long, Jingyang Li, Haiyang Yu, Jian Sun & Yongbin Li. 2022. A survey on neural open information extraction: Current status and future directions. arXiv [cs.CL]. 10.48550/arXiv.2205.11725

Proposta recebida em Outubro 2024 e aceite para publicação em Dezembro 2024.

RoBERTaLexPT: um modelo RoBERTa jurídico pré-treinado com deduplicação para língua Portuguesa

RoBERTaLexPT: A Legal RoBERTa Model pretrained with deduplication for Portuguese

Eduardo Garcia
Universidade Federal de Goiás

Nádia Silva
Universidade Federal de Goiás

Juliana Gomes
Universidade Federal de Goiás

Hidelberg Albuquerque
Universidade Federal Rural de Pernambuco

Ellen Souza
Universidade Federal Rural de Pernambuco

Felipe Siqueira
Universidade de São Paulo

Eliomar Lima
Universidade Federal de Goiás

André de Carvalho
Universidade de São Paulo

Resumo

Este trabalho investiga a aplicação do Processamento de Linguagem Natural (PLN) no contexto jurídico para a língua portuguesa, enfatizando a importância de adaptar modelos pré-treinados, como o RoBERTa, a partir de *corpora* especializados no domínio jurídico. Compilamos e pré-processamos um *corpus* jurídico em português, o *corpus* "LegalPT," abordando os desafios da alta duplicação de documentos em *corpora* jurídicos e medindo o impacto dos hiperparâmetros e da inicialização de *embeddings*. Experimentos revelaram que o pré-treinamento em dados jurídicos e em dados gerais resultou em modelos mais eficazes para tarefas jurídicas, com o nosso modelo, intitulado RoBERTaLexPT, superando modelos maiores treinados em *corpora* genéricos e outros modelos jurídicos de trabalhos relacionados. Também agregamos um *benchmark* jurídico, o *benchmark* "PortuLex." Este estudo contribui para melhorar as soluções de PLN no contexto jurídico brasileiro, fornecendo modelos aprimorados, um *corpus* especializado e um conjunto de dados de referência. Para fins de reprodutibilidade, disponibilizaremos o código, os dados e os modelos relacionados.

Palavras chave

modelo de linguagem; domínio jurídico; *benchmark*

Abstract

This work investigates the application of Natural Language Processing (NLP) in the legal context for the Portuguese language, emphasizing the importance of adapting pre-trained models, such as RoBERTa, from specialized corpora in the legal domain. We compiled and pre-processed a Portuguese Legal corpus, LegalPT corpus, addressing challenges of high document duplication in legal corpora, and measuring the impact of hyperparameters and embedding initialization. Experiments revealed that pre-training on legal and general data resulted in more effective models for legal tasks, with RoBERTaLexPT outperforming larger models trained on generic corpora, and other legal models from related works. We also aggregated a legal benchmark, PortuLex benchmark. This study contributes to improving NLP solutions in the Brazilian legal context, providing enhanced models, a specialized corpus, and a benchmark dataset. For reproducibility, we will make related code, data, and models available.

Keywords

language model; legal domain; benchmark

1. Introdução

Profissionais e pesquisadores da área jurídica enfrentam, diariamente, a necessidade de lidar com um volume substancial de textos legais que abrange legislação, jurisprudência, contratos e petições, com seus subdomínios específicos (e.g., processual, penal, trabalhista, previdenciário) e suas esferas de atuação (e.g., tipos de tribunais, jurisdição) (Akbik et al., 2016; Firdaus Solihin & Makarim, 2021; Cabrera-Diego & Gheewala, 2023). A natureza desses textos adiciona desafios significativos às aplicações de Processamento de Linguagem Natural (PLN), uma vez que textos legais são tipicamente, extensos, desestruturados, apresentam ruídos e são redigidos utilizando jargões e uma linguagem técnica específica do domínio (Kapoor et al., 2022).

O número de aplicações de PLN no campo jurídico tem crescido de maneira expressiva, impulsionado, sobretudo, pelos avanços em métodos especializados capazes de lidar com a comple-

DOI: 10.21814/lm.16.2.457
This work is Licensed under a Creative Commons Attribution 4.0 License

xidade inerente à linguagem jurídica (Zhong et al., 2020). Modelos de linguagem pré-treinados, como o *Bidirectional Encoder Representations from Transformers (BERT)* (Devlin et al., 2019), têm apresentado resultados significativos em várias tarefas de PLN voltadas para a língua portuguesa (Souza et al., 2020; Costa et al., 2022). Estudos indicam que o desempenho dos modelos pode ser aprimorado quando são pré-treinados em *corpora* específicos de um determinado domínio, como textos jurídicos (Chalkidis et al., 2020). Esse processo, conhecido como adaptação de domínio, permite que os modelos se adéquem melhor às características e ao vocabulário especializado de cada área.

No que diz respeito à adaptação de domínio, estudos anteriores têm se concentrado em investigar o impacto da seleção de dados por meio de técnicas básicas de deduplicação, ou seja, remoção de duplicatas (Lee et al., 2022; Tirumala et al., 2023). A deduplicação reduz o viés nos modelos, melhora a generalização e o desempenho, e mitiga o risco de *overfitting* ao evitar a memorização de padrões repetitivos, fortalecendo sua robustez (Brown et al., 2020). Essas técnicas oferecem aos pesquisadores uma abordagem de baixo risco para aprimorar o desempenho dos modelos de linguagem, mesmo ao simplesmente adicionar mais dados, ainda que não necessariamente inéditos. No contexto da língua portuguesa, estudos recentes têm apresentado resultados promissores ao desenvolver modelos específicos para textos legais em português (Polo et al., 2021; Viegas et al., 2023). Entretanto, tais pesquisas têm focado predominantemente em tarefas isoladas de PLN aplicadas ao domínio jurídico, o que limita a avaliação dos benefícios proporcionados pela adaptação de domínio nesses modelos e dificulta a realização de comparações entre eles.

Em Garcia et al. (2024), disponibilizamos (i) o *corpus* "LegalPT", que trata dos desafios relacionados à alta duplicação de documentos em *corpora* jurídicos, (ii) o "PortuLex", um *benchmark* jurídico em português composto por tarefas de NER e classificação, e (iii) o modelo "RoBERTaLexPT", treinado com a arquitetura RoBERTa (Liu et al., 2019) sobre os *corpora* LegalPT e "CrawlPT", superando o desempenho dos modelos jurídicos disponíveis para a língua portuguesa. Nesta versão estendida, as principais contribuições deste estudo incluem: (iv) a disponibilização de uma nova versão do RoBERTaLexPT, aprimorado com um pré-treinamento adicional utilizando textos específicos do domínio; (v) a análise do impacto de realizar *further pre-training*[1] em um modelo já existente em comparação com iniciar um pré-treinamento do zero; e (vi) a avaliação de modelos jurídicos e *benchmarks* genéricos, mensurando o impacto da transferência de aprendizado.

Este artigo está estruturado da seguinte forma: a Seção 2 fornece uma visão geral dos trabalhos relacionados com foco em modelos jurídicos pré-treinados e técnicas de deduplicação de *corpus*. Na Seção 3, são apresentados os *corpora* utilizados nos dados pré-treinados e o *benchmark* PortuLex, bem como uma análise comparativa das taxas de deduplicação. A Seção 4 apresenta o método utilizado para pré-treinamento e ajuste fino dos modelos. A Seção 5 apresenta e discute os resultados. A Seção 6 apresenta as contribuições, limitações e oportunidades de pesquisa. Por fim, o Apêndice A apresenta os melhores resultados da busca de hiperparâmetros utilizados no pré-treinamento dos modelos.

2. Trabalhos relacionados

A aquisição de uma quantidade massiva de novos dados é essencial para alcançar um desempenho ótimo em modelos de linguagem. Como regra geral, quanto maior o número de documentos utilizados, maior será a chance de melhoria do desempenho dos modelos em tarefas de PLN (Kaplan et al., 2020). Estudos empíricos têm demonstrado consistentemente que a adaptação de modelos baseados no modelo Transformer como o BERT, para *corpora* específicos de domínio (Chalkidis et al., 2020; Lee et al., 2019; Beltagy et al., 2019) pode resultar em melhorias substanciais de desempenho.

Através do pré-treinamento em textos jurídicos de uma legislação, um modelo pode aprender capacidades jurídicas específicas de cada país (Paul et al., 2023). Trabalhos em idiomas como chinês (Xiao et al., 2021), italiano (Licari & Comandè, 2022), romeno (Masala et al., 2021), espanhol (Gutiérrez-Fandiño et al., 2021), árabe (Al-qurishi et al., 2022) e francês (Douka et al., 2021), revelaram que os modelos jurídicos superam seus equivalentes de domínio geral em cerca de até 5%, particularmente quando seus dados de treinamento estão estreitamente alinhados.

Modelos de linguagem jurídica em português, como o "BERTikal" (Polo et al., 2021) e o "JurisBERT" (Viegas et al., 2023), relataram desempenho superiores em uma tarefa jurídica específica

[1] *Further pre-training* ou Pré-treinamento continuado é uma técnica que faz um "pré-treinamento adicional" com dados específicos do domínio.

Corpus **Domínio**	**Recursos**	**Documentos (M)**	***Tokens*** **(B)**	**Tamanho (GiB)**
LegalPT / Jurídico	*MultiLegalPile (PT)* (Niklaus et al., 2024)	17,7	15,7	88,1
	Ulysses-Tesemô (Siqueira et al., 2024)	2,2	4,9	27
	ParlamentoPT (Rodrigues et al., 2023)	2,7	0,5	2,5
	Iudicium Textum (Willian Sousa & Fabro, 2019)	0,2	0,1	0,8
	Acórdãos TCU (Bonifacio et al., 2020)	0,6	0,6	3,2
	DataSTF (2019)[1]	0,7	0,6	3,4
	Total	**24,2**	**22,5**	**125,1**
CrawlPT / Geral	brWaC (Wagner Filho et al., 2018)	3,5	3,1	16,3
	CC100 (PT) (Conneau et al., 2020)	39	9,4	49,1
	OSCAR-2301 (PT) (Abadji et al., 2022)	18	18,1	97,8
	Total	**60,5**	**30,6**	**163,3**

[1] https://legalhackersnatal.wordpress.com/2019/05/09/mais-dados-juridicos/

Tabela 1: Tamanho dos *corpora* em termos de milhões de documentos, bilhões de *tokens* e tamanho de arquivo em GiB.

em comparação com o "BERTimbau" (Souza et al., 2020), um modelo de linguagem genérico em português. Por sua vez, na pesquisa de Niklaus et al. (2024), foi realizado treinamento tanto em modelos jurídicos multilíngues quanto em modelos monolíngues, incluindo o português, com uma quantidade substancial de dados. Apesar disso, o modelo monolíngue em português não conseguiu superar o desempenho do BERTimbau em múltiplas tarefas jurídicas.

No desenvolvimento de *corpora* extensos, como "MC4" (Xue et al., 2021), "CC100" (Conneau et al., 2020) e "brWaC" (Wagner Filho et al., 2018), é comum o emprego de técnicas que removem documentos duplicados. Esse processo visa aumentar a qualidade dos dados e prevenir vieses indesejados durante o treinamento de modelos de aprendizado de máquina. No entanto, entre os *corpora* jurídicos em português examinados neste estudo (Willian Sousa & Fabro, 2019; Bonifacio et al., 2020; Niklaus et al., 2024), não foram encontradas informações sobre o uso de técnicas de deduplicação.

Lee et al. (2022) demonstram que conjuntos de dados deduplicados tendem a melhorar o desempenho de modelos de linguagem causal. Modelos treinados em conjuntos de dados com tendências de duplicação podem memorizar os dados, potencialmente levando à contaminação entre as divisões de treinamento e validação. Nossa hipótese é que essa diferença de desempenho também pode ser observada em modelos de linguagem mascarada.

Nossa pesquisa é semelhante à de Beltagy et al. (2019), Chalkidis et al. (2020), e Lee et al. (2019) quanto ao pré-treinamento de modelos BERT para o domínio. Seguimos principalmente as diretrizes de treinamento de Liu et al. (2019), aplicamos a deduplicação de texto conforme descrito em Lee et al. (2022), e nos concentramos obter *corpora* jurídicos nos idiomas português brasileiro e europeu. Ao combinar contribuições de cada um desses trabalhos, visamos preencher as lacunas nos modelos de português de última geração adaptados ao domínio jurídico. Até onde sabemos, nosso trabalho também é o primeiro a propor um *benchmark* adaptado a este domínio.

3. *Corpora*

Para esta pesquisa, buscamos adquirir o maior número possível de dados no domínio jurídico para a língua portuguesa. Desta forma, foram compilados dois *corpora* principais para o pré-treinamento: "LegalPT", um *corpus* específico para o domínio jurídico, e "CrawlPT", um *corpus* de uso geral usado para comparação. Foram utilizados unicamente recursos públicos disponíveis na internet, garantindo acessibilidade e transparência no processo de extração dos dados. Adicionalmente, criamos o *benchmark* "PortuLex", composto por um conjunto de tarefas supervisionadas jurídicas projetadas para avaliação dos modelos de linguagem desenvolvidos. A Tabela 1, resume os *corpora* utilizados neste estudo, detalhados a seguir.

3.1. *Corpus* LegalPT

Os textos jurídicos a seguir foram agregados para compor o *corpus* LegalPT, utilizado no pré-treinamento dos modelos desta pesquisa:

Corpus	Documentos	Tarefa	Instâncias	Treinamento	Validação	Teste
RRI (Aragy et al., 2021)	70	CLS	10,78 mil	8,26 mil	1,05 mil	1,47 mil
LeNER-Br (Luz de Araujo et al., 2018)	70	NER	10,4 mil	7,83 mil	1,18 mil	1,39 mil
UlyssesNER-Br—PL-Corpus (Albuquerque et al., 2022)	154	NER	4,30 mil	3,28 mil	489	524
FGV-STF (Correia et al., 2022)	594	NER	594	415	60	119
Total	888	—	26,074 mil	19,785 mil	2,779 mil	3,503 mil

Tabela 2: *Benchmark* PortuLex usado nas tarefas de classificação de sentenças (CLS) e de *tokens* com NER.

MultiLegalPile (Niklaus et al., 2024)[2] é um *corpus* multilíngue de textos jurídicos que compreende 689 GiB de dados, abrangendo 24 idiomas em 17 jurisdições. O *corpus* é separado por idioma, e o subconjunto em português contém 88.1 GiB de dados e 15,7 bilhões de palavras. Este subconjunto inclui jurisprudências do Tribunal de Justiça de São Paulo, recursos do 5º Tribunal Regional Federal (Menezes-Neto & Clementino, 2022) (BRCAD-5), o subconjunto português de documentos jurídicos da União Europeia (EUR-Lex)[3], e um filtro para documentos jurídicos do MC4 (Xue et al., 2021).

Ulysses-Tesemõ (Siqueira et al., 2024)[4] é um *corpus* jurídico em português brasileiro, composto por 2,2 milhões de documentos, totalizando cerca de 27 GiB de texto obtidos de 96 fontes de dados diferentes. Essas fontes abrangem documentos jurídicos, legislativos, artigos acadêmicos, notícias e comentários relacionados. Os dados foram coletados por meio de *web scraping* de sites governamentais.

ParlamentoPT (Rodrigues et al., 2023)[5] é um *corpus* jurídico utilizado no treinamento de modelos de linguagem em português europeu. Os dados foram coletados do portal do governo português, e consiste em 2,7 milhões de documentos de transcrições de debates no Parlamento português.

Iudicium Textum (Willian Sousa & Fabro, 2019)[6] consiste em acórdãos, votos e relatórios do Supremo Tribunal Federal (STF) do Brasil, publicados entre 2010 e 2018. O conjunto de dados contém 0.8 GiB de dados extraídos de PDFs.

Acórdãos TCU (Bonifacio et al., 2020)[7] é um conjunto de dados aberto do Tribunal de Contas da União, contendo 600.000 documentos obtidos por *web scraping* de *sites* governamentais. Os documentos abrangem o período de 1992 a 2019.

DataSTF[8] é um conjunto de dados de decisões monocráticas do Superior Tribunal de Justiça do Brasil, contendo 700.000 documentos (3.4 GiB de dados).

3.2. *Corpus* CrawlPT

Para avaliar o impacto da deduplicação e do tamanho dos dados em comparação com outros modelos de linguagem gerais em português, também aplicamos o mesmo processo aos seguintes *corpora* gerais em português, para formação do *corpus* "CrawlPT":

brWaC (Wagner Filho et al., 2018)[9] é um *corpus* extenso em português brasileiro, contendo cerca de 3,5 milhões de documentos e 3,31 bilhões de tokens, extraídos de mais de 120 mil websites diferentes. Construído seguindo a metodologia *WaCky*, o *corpus* utilizou técnicas de deduplicação, garantindo alta diversidade e qualidade dos dados.

CC100 (Conneau et al., 2020)[10] é um *corpus* criado para treinar o modelo Transformer multilíngue XLM-R. O *corpus* contém 2 TB

[2] https://huggingface.co/datasets/joelniklaus/Multi_Legal_Pile
[3] https://eur-lex.europa.eu/homepage.html
[4] https:github.com/ulysses-camara/ulysses-tesemo
[5] https://huggingface.co/datasets/PORTULAN/parlamento-pt
[6] https://dadosabertos.c3sl.ufpr.br/acordaos/
[7] https://www.kaggle.com/datasets/ferraz/acordaos-tcu
[9] https://www.inf.ufrgs.br/pln/wiki/index.php?title=BrWaC
[10] https://github.com/facebookresearch/cc_net

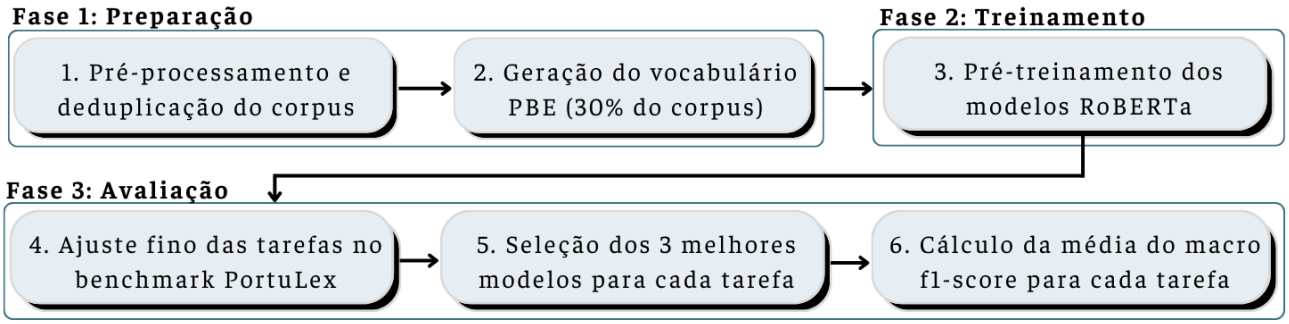


Figura 1: Fluxograma para pré-treino e avaliação no *benchmark* PortuLex.

de dados limpos das capturas realizadas no período de janeiro a dezembro de 2018 do projeto *Common Crawl* em 100 idiomas. Para esta pesquisa, foi utilizado o subconjunto em português do CC-100, que contém 49,1 GiB de texto e devido a sua natureza pode incluir todas as variantes do português disponíveis na *World Wide Web* como o brasileiro, europeu e africano.

OSCAR-2301 (Abadji et al., 2022)[11] é um *corpus* multilíngue extraído do *dump* de novembro a dezembro de 2022 do Common Crawl. Esta versão do OSCAR foi projetada para atender a uma ampla gama de tarefas de PLN e aprendizado de máquina, com dados organizados por idiomas e categorias temáticas. Utilizamos o subconjunto em português do OSCAR-2301, que contém 97,8 GiB de texto e também pode incluir diversas variantes do idioma português.

3.3. *Benchmark* PortuLex

Nossa pesquisa concentra-se na aquisição de dados de treinamento supervisionado abertos, anotados por especialistas jurídicos. Para manter a alta qualidade do *benchmark*, evitamos deliberadamente conjuntos de dados gerados automaticamente. À luz desses esforços, apresentamos o *benchmark* "PortuLex", composto por quatro tarefas de classificação, sendo uma para Classificação de Sentenças (CLS) e três para Classificação de Sentenças de Tokens utilizando NER, projetadas para avaliar a qualidade e o desempenho de modelos de linguagem no domínio jurídico português. A composição do PortuLex é mostrada na Tabela 2. O Portulex é composto pelos seguintes *corpora*:

RRI Identificação de Papéis Retóricos (*Rhetorical Role Identification*)(Aragy et al., 2021)[12] é um conjunto de dados de anotações retóricas no domínio jurídico, focando em sentenças extraídas de decisões judiciais do Tribunal de Justiça de Mato Grosso do Sul (Brasil). O *corpus* abrange 70 petições iniciais, contendo aproximadamente 10.000 sentenças rotuladas manualmente. São definidos oito papéis retóricos em alinhamento com o Código de Processo Civil Brasileiro, incluindo a identificação de partes, fatos, argumentos, fundamento jurídico, jurisprudência, pedidos, valor da causa e "outros".

LeNER-Br (Luz de Araujo et al., 2018)[13] é o primeiro *corpus* de NER para o domínio jurídico em português brasileiro. Ele compreende 66 documentos provenientes de tribunais superiores e estaduais, além de 04 textos de leis, totalizando 70 documentos jurídicos anotados com seis classes de entidades, sendo quatro universais (organização, pessoa, tempo e localização) e duas do domínio (legislação e jurisprudência).

UlyssesNER-Br (Albuquerque et al., 2022)[14] é um *corpus* de documentos legislativos brasileiros para NER, composto por 154 projetos de lei (PL-corpus) e 800 consultas legislativas internas (ST-corpus) da Câmara dos Deputados do Brasil. Por ter sido composto de projetos de lei públicos, somente o PL-corpus foi disponibilizado. O *corpus* abrange dois níveis de granularidade, por categorias e por tipos, com 18 tipos de entidades anotadas manualmente e estruturadas em 7 classes semânticas.

[11] https://huggingface.co/datasets/oscar-corpus/OSCAR-2301

[12] https://bit.ly/rhetoricalrole

[13] https://huggingface.co/datasets/peluz/lener_br

[14] https://github.com/ulysses-camara/ulysses-ner-br

FGV-STF (Correia et al., 2022)[15] é um *corpus* de documentos jurídicos para extração de entidades nomeadas, composto por 594 decisões do STF, selecionadas manualmente por especialistas do domínio entre 2009 e 2018. Os dados são anotados com diferentes níveis de granularidade, com foco principal em fundamento jurídico. Essas classes abrangem precedentes, citações acadêmicas e referências legislativas, com cada categoria contendo subtipos mais específicos de entidades. Usamos apenas as quatro principais entidades de granularidade mais ampla.

4. Método

Esta seção detalha o método utilizado, abrangendo a arquitetura do modelo, o treinamento, os conjuntos de dados e a avaliação. Conforme a Figura 1, o processo foi dividido em três fases: preparação dos *corpora*, treinamento e avaliação dos modelos de linguagem, que serão descritas a seguir.

4.1. Fase 1: Preparação

Nesta fase, os *corpora* foram inicialmente pré-processados para verificação de duplicatas. Seguindo a abordagem de deduplicação de Lee et al. (2022), analisamos os subconjuntos do *corpus* LegalPT usando os algoritmos *MinHash* (Broder, 2000) e *Locality Sensitive Hashing* (Har-Peled et al., 2012), identificando agrupamentos de documentos duplicados. Para isso, utilizamos duas técnicas que facilitam a comparação entre grandes volumes de dados: a técnica de *n-gramas* com sequências de cinco palavras consecutivas (5-gramas), para captura de padrões textuais, e uma assinatura de 256 elementos, para representar um resumo compacto do documento analisado, facilitando a comparação entre textos. Consideramos dois documentos idênticos quando sua Similaridade *Jaccard* (Jaccard, 1912) exceder 0.7. Quando encontradas, as duplicatas são removidas do conjunto de treinamento para melhorar a eficiência e a generalização do modelo. Os resultados do processo de deduplicação para os subconjuntos dos *corpora* LegalPT e CrawlPT podem ser encontrados na Tabela 3.

Após o passo de pré-processamento e deduplicação, buscando evitar que os modelos de domínio sejam limitados por um vocabulário genérico, aplicamos a biblioteca *HuggingFace Tokenizers*[16] e o algoritmo *Byte-Pair Encoding (BPE)* para treinar um vocabulário específico para cada *corpus* do pré-treinamento.

4.2. Fase 2: Treinamento

Nesta fase, foi executado o processo de pré-treinamento do nosso modelo de linguagem jurídica. Foi utilizado o modelo RoBERTa$_{base}$, um modelo de linguagem baseado no modelo Transformer (Liu et al., 2019). Nosso modelo foi pré-treinado com quatro configurações distintas: (i) exclusivamente no *corpus* LegalPT (RoBERTaLegalPT$_{base}$), (ii) exclusivamente no *corpus* CrawlPT (RoBERTaCrawlPT$_{base}$), (iii) combinando ambos os *corpora* (RoBERTaLexPT$_{base}$) e, (iv) exclusivamente no brWaC (RoBERTaTimbau$_{base}$).

O pré-treinamento envolveu 62.500 etapas, com um *batch size* de 2048 sequências, cada uma contendo no máximo 512 *tokens*. Esta configuração é semelhante à utilizada pelo BERTimbau, expondo o modelo a aproximadamente 65 bilhões de *tokens* durante o treinamento. Adotamos a configuração padrão dos hiperparâmetros do modelo RoBERTa. Durante o pré-treinamento, adotamos a técnica de modelagem de linguagem mascarada (*Masked Language Modeling — MLM*), onde 15% dos *tokens* de entrada foram mascarados aleatoriamente, e o modelo previu os *tokens* mascarados com base no contexto. A otimização foi conduzida com o otimizador *AdamW*, utilizando um escalonamento linear seguido de uma taxa de aprendizado com decaimento linear. A Tabela 4 apresenta os hiperparâmetros utilizados.

O processo de pré-treinamento foi executado em um cluster DGX-A100, composto de 2 GPUs Nvidia A100 com 80 GB de memória. Utilizamos a biblioteca *Fairseq* (Ott et al., 2019)[17], desenvolvida para o treinamento de modelos de linguagem usando redes neurais, entre outras tarefas. O treinamento completo de uma única configuração levou aproximadamente três dias.

4.3. Fase 3: Avaliação

A fase de Avaliação do modelo é composta por três tarefas principais: ajuste fino, seleção dos melhores modelos, e cálculo do F1-score médio para comparação dos resultados.

Inicialmente, realizou-se o ajuste fino dos modelos treinados nos conjuntos de dados do PortuLex, seguindo a abordagem proposta por

[15] O acesso ao *corpus* é disponibilizado mediante solicitação direta ao autor do artigo.

[16] https://github.com/huggingface/tokenizers

[17] https://github.com/facebookresearch/fairseq

Corpus	**Sub-*corpus***	**Docs.**	**Docs. depois da deduplicação**	**Duplicatas (%)**
LegalPT	Ulysses-Tesemõ	2.216.656	1.737.720	21,61
	MultiLegalPile (PT)			
	- CJPG	14.068.634	6.260.096	55,50
	- BRCAD-5	3.128.292	542.680	82,65
	- EUR-Lex (Caselaw)	104.312	78.893	24,37
	- EUR-Lex (Contracts)	11.58	8.511	26,51
	- EUR-Lex (Legislation)	232.556	95.024	59,14
	- Legal MC4	191.174	187.637	1,85
	ParlamentoPT	2.670.846	2.109.931	21,00
	Iudicium Textum	198.387	153.373	22,69
	Acordãos TCU	634.711	462.031	27,21
	DataSTF	737.769	310.119	57,97
	Total	**24.194.918**	**11.946.015**	**50,63**
CrawlPT	brWaC	3.530.796	3.513.588	0,49
	OSCAR-2301 (PT)	18.031.400	10.888.966	39,61
	CC100 (PT)	38.999.388	38.059.979	2,41
	Total	**60.561.584**	**52.462.533**	**13,37**

Tabela 3: Taxa de duplicação encontrada com algoritmo Minhash-LSH (Lee et al., 2022).

Hiperparâmetros	**RoBERTa**$_{base}$
Number of layers	12
Hidden size	768
FFN inner hidden size	3072
Attention heads	12
Attention head size	64
Dropout	0.1
Attention dropout	0.1
Warmup steps	6k
Peak learning rate	7e-4
Batch size	2048
Weight decay	0.01
Maximum training steps	62.5k
Learning rate decay	Linear
AdamW ϵ	1e-6
AdamW β_1	0.9
AdamW β_2	0.98
Gradient clipping	0.0

Tabela 4: Hiperparâmetros utilizados no pré-treinamento do modelo RoBERTa.

Devlin et al. (2019). Esse método treina um codificador Transformer bidirecional para tarefas de classificação de texto e reconhecimento de entidades nomeadas. A Tabela 5 apresenta o espaço de busca explorado durante o *grid search*. Os hiperparâmetros ajustados incluíram o tamanho do lote (*batch size*) e a taxa de aprendizado (*learning rate*), enquanto os demais foram mantidos constantes.

Após a etapa de ajuste fino, realizamos a avaliação dos modelos com base nos 3 melhores *checkpoints* identificados no conjunto de validação. Por fim, a métrica final foi obtida calculando a média aritmética do macro F1-score nas divisões de teste do conjunto de dados. Essa abordagem permite avaliar a robustez dos modelos, minimizando o risco de *overfitting* nos dados de treinamento. Ao selecionar múltiplos *checkpoints*, garantimos uma avaliação mais equilibrada e robusta, com a expectativa de que os modelos mantenham um bom desempenho em dados inéditos e generalizem de forma eficiente.

Hiperparâmetros	**Espaço de busca**
Batch size	{16, 32}
Learning rate	{7.5e-6, 1e-5, 2.5e-5, 5e-5}
Dropout of task layer	0.0
Warmup steps	100
Weight decay	0.01
Maximum training epochs	50
Learning rate scheduler	Constant
Optimizer	AdamW
AdamW ϵ	1e-8
AdamW β_1	0.9
AdamW β_2	0.999
Early stopping patience	750 steps
Early stopping threshold	0.001 (F1-score)

Tabela 5: Espaço de busca de hiperparâmetros para modelos de ajuste fino treinados no *benchmark* PortuLex.

Modelo	Hiperparâmetros					PortuLex *Score* (%)
	Tamanho do lote	Taxa de aprendizado	Inicialização de pesos	Passos	Épocas	
BERTimbau$_{base}$	128	1e-4	mBERT (sem *embeddings*)	1,000,000	8	83.78
RoBERTaTimbau$_{base}$	2048	1e-4	XLM-R$_{base}$ (sem *embeddings*)	30,000	8	84.01*
				62,500	17	83.96*
	2048	7e-4	XLM-R$_{base}$ (sem *embeddings*)	30,000	8	83.40
				62,500	17	83.94*
Corpus: brWaC (16GiB)	2048	7e-4	Aleatória	30,000	8	83.36
				62,500	17	**84.29***

Tabela 6: Macro F1-*Score* no *benchmark* PortuLex para modelos RoBERTa$_{base}$ em português pré-treinados no brWaC. As pontuações que superaram o BERTimbau$_{base}$ estão marcadas com asterisco, e a maior pontuação em negrito.

5. Resultados e Discussão

Esta seção apresenta os resultados dos experimentos com modelos baseados no RoBERTa e o RoBERTaLexPT, pré-treinado em um *corpus* combinado de textos jurídicos e genéricos. Avaliamos o impacto dos hiperparâmetros e a combinação de *corpora* (Seções 5.1 e 5.2), exploramos o pré-treinamento adicional em domínios específicos (Seção 5.3) e analisamos o aumento de parâmetros e tempo de treinamento (Seção 5.4). Por fim, comparamos nosso modelo com modelos jurídicos existentes e avaliamos sua versatilidade em *benchmarks* genéricos (Seções 5.5 e 5.6).

5.1. Replicando o BERTimbau com RoBERTa

Os experimentos nesta seção têm como objetivo investigar como diferentes hiperparâmetros afetam o desempenho do modelo em comparação com o RoBERTa, utilizando um *batch size* maior.

O modelo BERTimbau (Souza et al., 2020) foi pré-treinado com um comprimento máximo de sequência de entrada variando entre 128 e 512 *tokens*, um vocabulário de 29.794 *tokens* treinado na Wikipedia em português (Wikipedia PT), com *batch size* de 128, e executado por 1 milhão de etapas (ou 8 épocas) no *corpus* brWaC. Durante esse processo, o modelo foi exposto a um total de 65 bilhões de *tokens*. Os pesos de treinamento foram inicializados a partir dos modelos mBERT$_{base}$ e BERT$_{large}$, removendo a camada de *embeddings* inicial para acomodar o novo vocabulário em língua portuguesa.

Nesta pesquisa, avaliamos variações na taxa de aprendizado (*learning rate*) do modelo, no número de épocas de treinamento e na inicialização. Os modelos foram baseados na arquitetura RoBERTa$_{base}$, com um comprimento de *tokenização* fixo de 512 tokens e um vocabulário BPE de 50.265 *tokens*, treinado na Wikipedia PT. Os *checkpoints* foram avaliados com base no *benchmark* PortuLex proposto neste trabalho. Os resultados estão resumidos na Tabela 6.

Para manter a comparabilidade de custos computacionais com o BERTimbau, estabelecemos um limite de 65 bilhões de *tokens* de treinamento. Com o novo vocabulário BPE, corresponde a aproximadamente 17 épocas no *corpus* brWaC, ou 62.500 etapas de treinamento com um tamanho de lote de 2048 e comprimento de tokenização de 512 *tokens*. Também reportamos os resultados para 8 épocas (equivalente a 30.000 etapas de treinamento em nossa configuração). Utilizamos o modelo pré-treinado XLM-R$_{base}$ (Conneau et al., 2020) como inicialização, descartando sua camada de representações vetoriais.

Utilizando essa inicialização, foi possível perceber que o modelo conseguiu superar o BERTimbau no *benchmark* PortuLex com apenas 30.000 etapas de treinamento, alcançando um macro F1-score médio de 84,01%, em comparação com 83,78% do BERTimbau. No entanto, a execução do treinamento por mais tempo ou o ajuste da taxa de aprendizado não pareceram melhorar o desempenho do modelo. Com inicialização aleatória, nosso modelo RoBERTa apresentou desempenho inferior ao BERTimbau no ponto de 8 épocas, mas superou a inicialização do XLM-R$_{base}$ ao treinar por um período mais longo. Com 17 épocas, o modelo atingiu um macro F1-score médio de 84,29% no *benchmark* PortuLex.

Esse comportamento também foi observado no gráfico de perda do MLM no subconjunto de validação, como mostrado na Figura 2. Entre 54.000 e 55.000 etapas de treinamento, o modelo sem inicialização superou o modelo inicializado com o XLM-R. Esse ponto de inflexão

sugere que, em determinado estágio, o modelo sem inicialização pode alcançar maior capacidade de generalização, apesar do início mais lento em comparação com o modelo pré-treinado. Isso reforça a importância de explorar diferentes abordagens de inicialização ao otimizar modelos de linguagem.

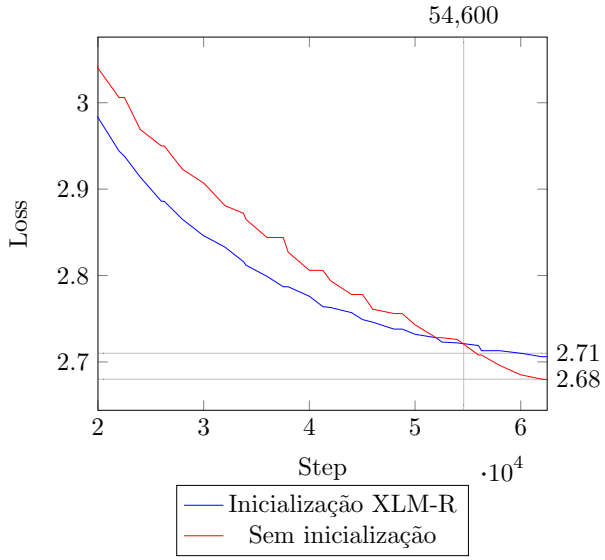


Figura 2: Perda do MLM no conjunto de validação dos modelos treinados no brWaC com diferentes inicializações.

Modelo	***Corpus***	**PortuLex *Score* (%)**
RoBERTaTimbau$_{base}$	brWaC	84,29
RoBERTaCrawlPT$_{base}$	CrawlPT	84,83
RoBERTaLegalPT$_{base}$	LegalPT	84,57
RoBERTaLexPT$_{base}$	LegalPT+CrawlPT	**85,41**

Tabela 7: Macro F1-*score* médio nos modelos pré-treinados no *benchmark* PortuLex. O RoBERTaLexPT$_{base}$, pré-treinado no *corpus* específico de domínio LegalPT e no *corpus* geral CrawlPT, alcança a maior pontuação.

5.2. Combinando *corpora* genéricos e jurídicos

Até o nosso conhecimento, as técnicas de adaptação de domínio ainda não exploraram se a combinação de um *corpus* específico de domínio com um *corpus* genérico pode melhorar o desempenho do modelo de aprendizagem, devido ao aumento do tamanho do *corpus* de pré-treinamento.

Para avaliar essa combinação, pré-treinamos modelos de linguagem no *corpus* CrawlPT (Seção 3.2), e na combinação de CrawlPT com LegalPT. Os modelos foram treinados com os hiperparâmetros definidos na Seção 5.1. O vocabulário BPE de cada modelo foi treinado com 30% dos documentos de seus respectivos *corpora*. A Tabela 7 apresenta o resultado da avaliação dos novos modelos no *benchmark* PortuLex, comparando-os com o modelo RoBERTaTimbau$_{base}$.

Curiosamente, quando usados individualmente para o pré-treinamento, o modelo CrawlPT apresentou desempenho superior ao LegalPT, apesar de seu domínio genérico. Mesmo com tamanhos semelhantes, o *corpus* CrawlPT possui mais dados únicos, com uma taxa de duplicação de 13,37%, em comparação com 50,63% no LegalPT. Este resultado pode indicar que um *corpus* genérico de alta qualidade pode ser tão eficiente quanto um *corpus* específico de domínio para o pré-treinamento de modelos de linguagem.

No entanto, ao combinar os dois *corpora*, o modelo RoBERTaLexPT resultante apresentou desempenho superior em comparação com os modelos pré-treinados em conjuntos de dados individuais. Esse resultado está de acordo com as conclusões de Kaplan et al. (2020) que afirmam que o tamanho do *corpus* é um fator chave para aumentar o desempenho do modelo, embora o estudo deles tenha examinado modelos de linguagem causais, diferindo dos modelos de linguagem mascarada utilizados em nossa pesquisa.

5.3. Pré-treinamento Adicional em Dados Específicos do Domínio

O pré-treinamento adicional (*further pretraining*) adapta modelos de linguagem previamente treinados a domínios específicos, aproveitando o conhecimento adquirido em *corpora* amplos e diversificados (Gururangan et al., 2020). Essa técnica combina a compreensão geral da linguagem com conhecimento específico do domínio, melhorando o desempenho em tarefas específicas, oferecendo vantagens computacionais ao reduzir o tempo e os recursos necessários, evitando o treinamento completo do zero.

Para explorar o impacto do pré-treinamento adicional, realizamos experimentos usando nosso modelo RoBERTaCrawlPT$_{base}$, que foi inicialmente pré-treinado no *corpus* CrawlPT. Continuamos seu pré-treinamento no *corpus* LegalPT por mais 30.000 passos, testando diferentes taxas de aprendizado. Os resultados desses experimentos são apresentados na Tabela 8.

Observamos que o pré-treinamento adicional do RoBERTaCrawlPT no *corpus* LegalPT por mais 30 mil passos melhorou seu desempenho no *benchmark* PortuLex. O melhor resultado foi alcançado com uma taxa de aprendizagem de 3e-4,

Modelo	**Passos**	***Corpus***	**Taxa de aprendizado**	**PortuLex Score (%)**
RoBERTaCrawlPT$_{base}$	62,5 mil	CrawlPT	7e-4	84,83
+ pré-treino continuado	+ 30 mil	+ LegalPT	7e-4	84,95
	+ 30 mil	+ LegalPT	3e-4	85,37
	+ 30 mil	+ LegalPT	1e-4	85,29
RoBERTaLexPT$_{base}$	62,5 mil	LegalPT + CrawlPT	7e-4	**85,41**

Tabela 8: Comparação dos resultados e um pré-treino adicional de domínio a partir de um modelo genérico e o RoBERTaLexPT$_{base}$, F1-*score* médio avaliados nas divisões de teste do benchmark PortuLex.

produzindo um F1-score médio de 85,37%, o que representa uma melhoria significativa em relação aos 84,83% do modelo inicial.

Curiosamente, essa abordagem de pré-treinamento adicional chega perto de igualar o desempenho do RoBERTaLexPT (85,41%), que foi pré-treinado no *corpus* combinado LegalPT+CrawlPT desde o início. A pequena diferença de desempenho (0,04%) sugere que ambas as estratégias (pré-treinamento combinado e pré-treinamento adicional) podem ser eficazes para o desenvolvimento de modelos de linguagem específicos para o domínio.

Os resultados destacam a importância do ajuste da taxa de aprendizagem em cenários de pré-treinamento adicional. Observamos que uma taxa de aprendizagem mais baixas (3e-4 e 1e-4) que a taxa de aprendizagem inicial de pré-treino do modelo (7e-4) tem um resultado melhor.

Embora o RoBERTaLexPT ainda mantenha uma leve vantagem no desempenho, a abordagem de pré-treinamento adicional oferece uma alternativa viável, especialmente quando os recursos computacionais são limitados ou quando se adaptam modelos existentes a novos domínios.

5.4. Expandindo Parâmetros e Tempo de Treinamento

Para investigar o impacto do tamanho do modelo e do tempo de treinamento no desempenho, conduzimos experimentos adicionais com variações do RoBERTaLexPT. Especificamente, treinamos uma versão do modelo base por um período mais longo (RoBERTaLexPT+$_{base}$) e implementamos uma versão maior do modelo (RoBERTaLexPT$_{large}$), também treinando-a por períodos curto e longo. A Tabela 9 apresenta os resultados desses experimentos.

Os resultados revelam que o aumento do tamanho do modelo de *base* para *large* não proporcionou benefícios substanciais para as tarefas avaliadas no *benchmark* PortuLex. O RoBERTaLexPT$_{large}$ com 62.5k passos de treinamento apresentou um desempenho ligeiramente inferior ao RoBERTaLexPT$_{base}$ (85.30% vs 85.41%). Quando treinados por 125k passos, ambos os modelos base e large atingiram o mesmo F1-score médio de 85.46%, indicando uma melhoria marginal em relação ao treinamento mais curto.

Esta falta de ganho significativo com o modelo maior pode ser atribuída principalmente ao tamanho limitado do *corpus* de treinamento. Com 125k passos, os modelos completam aproximadamente 2,5 épocas no dataset combinado LegalPT+CrawlPT. O fenômeno observado está alinhado com as descobertas de Kaplan et al. (2020) que demonstraram que o desempenho dos modelos de linguagem está intrinsecamente ligado à quantidade de dados de treinamento. No nosso caso, o modelo *large*, com sua maior capacidade, provavelmente requer um *corpus* substancialmente maior para demonstrar vantagens significativas sobre o modelo base.

Além disso, o RoBERTaLexPT$_{base}$ demonstrou notável eficiência, alcançando um desempenho competitivo com significativamente menos parâmetros (125M vs 355M) e em menos tempo de treinamento. Isso sugere que, para as tarefas específicas do PortuLex, o modelo base já possui capacidade suficiente para capturar as nuances necessárias.

Estes resultados destacam a importância de considerar cuidadosamente o equilíbrio entre o tamanho do modelo, o volume de dados de treinamento e a complexidade das tarefas alvo. Para o *benchmark* PortuLex e o *corpus* utilizado, o RoBERTaLexPT$_{base}$ parece oferecer o melhor equilíbrio entre eficiência computacional e desempenho.

Modelo	**Parâmetros (M)**	**Passos (mil)**	**Taxa de aprendizado**	**PortuLex) *Score* (%)**
RoBERTaLexPT$_{base}$	125	62,5	7e-4	85,41
RoBERTaLexPT+$_{base}$	125	125	7e-4	**85,46**
RoBERTaLexPT$_{large}$	355	62,5	2e-4	85,30
RoBERTaLexPT+$_{large}$	355	125	2e-4	**85,46**

Tabela 9: Comparação de desempenho entre modelos RoBERTaLexPT de diferentes tamanhos e tempos de treinamento, com F1-*score* médio nas divisões de teste do *benchmark* PortuLex. O símbolo **"+"** indica treinamento estendido.

Modelo	**LeNER**	**UlyNER-PL** Categorias/Tipos	**FGV-STF** Amplo	**RRIP**	**Média (%)**
BERTimbau$_{base}$	88,34	86,39/83,83	79,34	82,34	83,78
BERTimbau$_{large}$	88,64	87,77/84,74	79,71	**83,79**	84,60
Albertina-PT-BR$_{base}$	89,26	86,35/84,63	79,30	81,16	83,80
Albertina-PT-BR$_{xlarge}$	90,09	88,36/**86,62**	79,94	82,79	85,08
BERTikal$_{base}$	83,68	79,21/75,70	77,73	81,11	79,99
JurisBERT$_{base}$	81,74	81,67/77,97	76,04	80,85	79,61
BERTimbauLAW$_{base}$ (Viegas et al., 2023)	84,90	87,11/84,42	79,78	82,35	83,20
Legal-XLM-R$_{base}$	87,48	83,49/83,16	79,79	82,35	83,24
Legal-XLM-R$_{large}$	88,39	84,65/84,55	79,36	81,66	83,50
Legal-RoBERTa-PT$_{large}$	87,96	88,32/84,83	79,57	81,98	84,02
RoBERTaTimbau$_{base}$	89,68	87,53/85,74	78,82	82,03	84,29
RoBERTaLegalPT$_{base}$	90,59	85,45/84,40	79,92	82,84	84,57
RoBERTaLexPT$_{base}$	90,73	**88,56**/86,03	80,40	83,22	85,41
RoBERTaLexPT+$_{base}$	**90,97**	88,49/86,16	80,12	83,45	**85,46**
RoBERTaLexPT+$_{large}$	90,61	88,30/86,16	**80,60**	83,41	**85,46**

Tabela 10: Média de Macro F1-*score* (%) para vários modelos avaliados nas divisões de teste do *benchmark* PortuLex.

5.5. Comparação com outros modelos jurídicos

A Tabela 10 apresenta o desempenho do modelo RoBERTaLexPT em comparação a modelos jurídicos abertos em português no conjunto de dados do *benchmark* PortuLex.

Destacamos inicialmente que, apesar de usar apenas a configuração *base*, o RoBERTaLexPT+$_{base}$ superou modelos significativamente maiores, como BERTimbau$_{large}$, Albertina-PT-BR$_{xlarge}$ (Rodrigues et al., 2023), e Legal-XLM-R$_{large}$ (Niklaus et al., 2024). Este resultado destaca a eficácia do RoBERTaLexPT, resultante do pré-treinamento em dados jurídicos e genéricos combinados.

O RoBERTaLexPT alcançou o melhor desempenho nos conjuntos LeNER e FGV-STF, mesmo quando comparado a modelos muito maiores. Comparando com o UlyssesNER-Br/Pl-corpus, o RoBERTaLexPT+$_{base}$ obteve resultados competitivos. O único conjunto em que o RoBERTaLexPT+$_{base}$ é superado é o RRI, onde o BERTimbau$_{large}$ tem uma pequena vantagem de 0,34% de F1-score.

Em contraste, alguns trabalhos anteriores afirmaram ter obtido desempenho superior ao BERTimbau em certas tarefas jurídicas (Polo et al., 2021; Viegas et al., 2023). Esses modelos, na verdade, apresentaram um desempenho inferior ao BERTimbau em nossos experimentos com o *benchmark* PortuLex. Por exemplo, o JurisBERT alcançou apenas um F1-score médio de 79,61%, comparado aos 83,78% do BERTimbau. Uma possível explicação para essa discrepância é que as avaliações originais foram limitadas a um único conjunto de dados, provavelmente favorecendo os dados de treinamento específicos do modelo.

Em resumo, o RoBERTaLexPT e suas variações demonstrou consistentemente uma alta eficácia em PLN jurídico, mesmo em seu tamanho base. Com dados suficientes de pré-treinamento, acreditamos que nosso modelo pode superar modelos sobreparametrizados. Os resultados destacam a importância de dados de treinamento diversos em vez do simples aumento da escala do modelo.

5.6. Avaliações em *benchmarks* genéricos

Para avaliar a versatilidade dos modelos desenvolvidos neste trabalho, realizamos experimentos adicionais em tarefas genéricas de PLN em português. Utilizamos os *benchmarks* ASSIN2 Real et al. (2020) e HAREM (Santos et al., 2006), seguindo a metodologia proposta por Souza et al. (2020). Estes *benchmarks* incluem tarefas de Similaridade Textual de Sentenças (STS), Reconhecimento de Inferência Textual (RTE) e Reconhecimento de Entidades Nomeadas (NER). A Tabela 11 apresenta os resultados comparativos.

Os resultados mostram que nossos modelos, especialmente o RoBERTaLexPT+$_{large}$, apresentam desempenho competitivo em tarefas genéricas de PLN, apesar de terem sido treinados com foco em dados jurídicos. O RoBERTaLexPT+$_{large}$ supera o BERTimbau$_{large}$ em todas as tarefas, exceto no ASSIN2 RTE, onde a diferença é marginal (89.71% vs 89.77%). Notavelmente, o RoBERTaLexPT+$_{large}$ alcança o melhor desempenho nas tarefas de NER do HAREM, tanto no cenário padrão quanto no seletivo.

É interessante observar que o RoBERTaLegalPT$_{base}$, treinado exclusivamente em dados jurídicos, apresenta um desempenho inferior nas tarefas genéricas, como esperado. No entanto, o RoBERTaLexPT+$_{base}$, que combina dados jurídicos e genéricos, consegue manter um desempenho competitivo, superando até mesmo o BERTimbau$_{base}$ e o Albertina-PT-BR$_{base}$ em algumas tarefas.

O Albertina-PT-BR$_{xlarge}$ ainda mantém a liderança na média geral, principalmente devido ao seu desempenho superior nas tarefas do ASSIN2. Isso pode ser atribuído ao seu tamanho significativamente maior (900 milhões de parâmetros) e ao seu treinamento com maiores recursos.

Estes resultados demonstram que a abordagem de combinar dados jurídicos e genéricos no pré-treinamento, como feito no RoBERTaLexPT, não apenas melhora o desempenho em tarefas jurídicas específicas, mas também mantém uma forte capacidade de generalização para tarefas de PLN mais amplas. Isso sugere que o modelo adquiriu uma compreensão robusta da língua portuguesa em geral, além de conhecimentos específicos do domínio jurídico.

6. Conclusão

Este trabalho apresenta o RoBERTaLexPT, um modelo de linguagem jurídica em português pré-treinado em um *corpus* combinado de textos jurídicos e gerais. Até o nosso conhecimento, ao longo do processo, criamos o maior *corpus* jurídico em português (LegalPT) agregando diversas fontes, obtendo melhorias significativas de desempenho por meio da deduplicação, além de introduzir o *benchmark* PortuLex para uma avaliação rigorosa dos modelos.

Demonstramos que utilizar outros modelos como inicialização de pesos para o pré-treinamento de modelos de linguagem pode melhorar o desempenho em cenários com recursos limitados, embora isso tenha um *trade-off* em configurações de treinamento mais longas.

Nossos resultados indicam que combinar um *corpus* específico de domínio (LegalPT) e um *corpus* genérico (CrawlPT) para o pré-treinamento oferece benefícios complementares. Apesar de seu tamanho compacto em comparação com modelos anteriores, o RoBERTaLexPT na configuração base demonstra eficácia de ponta em PLN jurídico em português, destacando a importância dos dados de pré-treinamento em relação à escala do modelo.

Investigações adicionais sobre o impacto do tamanho do modelo e do tempo de treinamento revelaram que, para as tarefas do PortuLex, o aumento do tamanho do modelo de *base* para *large* não proporcionou benefícios substanciais. Isso sugere que o tamanho do *corpus* de treinamento pode ser um fator limitante mais significativo do que o tamanho do modelo para melhorias de desempenho em nosso contexto específico.

Além disso, avaliamos a versatilidade do RoBERTaLexPT em tarefas genéricas de PLN utilizando os *benchmarks* ASSIN2 e HAREM. Os resultados demonstraram que nosso modelo mantém um desempenho competitivo em tarefas não jurídicas, superando o BERTimbau em várias métricas. Isso indica que a abordagem de combinar dados jurídicos e genéricos no pré-treinamento não apenas melhora o desempenho em tarefas jurídicas específicas, mas também mantém uma forte capacidade de generalização.

O RoBERTaLexPT, juntamente com o LegalPT e o PortuLex, representam um avanço sig-

Modelo	**ASSIN2 RTE** (F1-*Score*)	**ASSIN2 STS** (Pearson)	**HAREM Padrão** (F1-*Score*)	**HAREM Seletivo** (F1-*Score*)	**Média (%)**
BERTimbau$_{base}$	88,76	84,15	74,02	80,53	81,87
BERTimbau$_{large}$	89,77	85,53	75,04	80,97	82,83
Albertina-PT-BR$_{base}$	86,91	82,25	73,72	78,82	80,43
Albertina-PT-BR$_{xlarge}$	**91,60**	**85,78**	76,62	82,75	**84,19**
RoBERTaTimbau$_{base}$	87,73	83,11	73,12	78,99	80,74
RoBERTaCrawlPT$_{base}$	88,28	83,20	74,51	80,24	81,56
RoBERTaLegalPT$_{base}$	85,34	78,89	67,73	75,25	76,80
RoBERTaLexPT+$_{base}$	88,68	83,19	73,97	81,36	81,80
RoBERTaLexPT+$_{large}$	89,71	85,50	**76,72**	**82,83**	83,69

Tabela 11: Comparação dos diversos modelos treinados neste trabalhos com modelos de linguagem genéricos em português nos *benchmarks* do ASSIN2 e HAREM.

nificativo para o processamento de linguagem natural do domínio jurídico em português, enfrentando limitações de recursos e modelos. Estes recursos oferecem uma base sólida para futuras pesquisas e aplicações no campo do PLN jurídico em português.

Trabalhos futuros podem explorar o pré-treinamento de modelos RoBERTa ainda maiores, investigar técnicas para melhor aproveitamento de modelos grandes com *corpora* limitados, expandir o *corpus* LegalPT, e aprimorar o *benchmark* PortuLex. Além disso, a aplicação do RoBERTaLexPT em tarefas jurídicas mais complexas e sua adaptação para outros domínios especializados são direções promissoras para pesquisas futuras.

Agradecimento

Este trabalho foi apoiado pelo Centro de Excelência em Inteligência Artificial (CEIA) do Instituto de Informática da Universidade Federal de Goiás (INF-UFG). Ellen Souza e Nadia Félix são financiadas pela FAPESP, por meio de um acordo entre a USP e a Câmara dos Deputados do Brasil. André C.P.L.F. de Carvalho é apoiado pelo CNPq. Agradecemos ao CEIA-UFG, ao Instituto de Inteligência Artificial (IAIA) e às agências de fomento à pesquisa, às quais expressamos nossa gratidão pelo suporte concedido ao desenvolvimento desta pesquisa.

Referências

Abadji, Julien, Pedro Ortiz Suarez, Laurent Romary & Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. arXiv [cs.CL]. 10.48550/arXiv.2201.06642

Akbik, Alan, Laura Chiticariu, Marina Danilevsky, Yonas Kbrom, Yunyao Li & Huaiyu Zhu. 2016. Multilingual information extraction with PolyglotIE. Em *26th International Conference on Computational Linguistics (COLING)*, 268–272.

Al-qurishi, Muhammad, Sarah Alqaseemi & Riad Souissi. 2022. AraLegal-BERT: A pretrained language model for Arabic legal text. Em *Natural Legal Language Processing Workshop (NLLP)*, 338–344. 10.18653/v1/2022.nllp-1.31

Albuquerque, Hidelberg O., Rosimeire Costa, Gabriel Silvestre, Ellen Souza, Nádia F. F. da Silva, Douglas Vitório, Gyovana Moriyama, Lucas Martins, Luiza Soezima, Augusto Nunes, Felipe Siqueira, João P. Tarrega, Joao V. Beinotti, Marcio Dias, Matheus Silva, Miguel Gardini, Vinicius Silva, André C. P. L. F. de Carvalho & Adriano L. I. Oliveira. 2022. UlyssesNER-Br: A corpus of Brazilian legislative documents for named entity recognition. Em *Computational Processing of the Portuguese Language (PROPOR)*, 3–14. 10.1007/978-3-030-98305-5_1

Aragy, Roberto, Eraldo Rezende Fernandes & Edson Norberto Caceres. 2021. Rhetorical role identification for Portuguese legal documents. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 557–571. 10.1007/978-3-030-91699-2_38

Beltagy, Iz, Kyle Lo & Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. Em *Conference on Empirical Methods in Natural Language Processing and*

the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 3615–3620. 10.18653/v1/D19-1371

Bonifacio, Luiz Henrique, Paulo Arantes Vilela, Gustavo Rocha Lobato & Eraldo Rezende Fernandes. 2020. A study on the impact of intradomain finetuning of deep language models for legal named entity recognition in Portuguese. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 648–662. 10.1007/978-3-030-61377-8_46

Broder, Andrei Z. 2000. Identifying and filtering near-duplicate documents. Em *Combinatorial Pattern Matching (CPM)*, 1–10. 10.1007/3-540-45123-4_1

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever & Dario Amodei. 2020. Language models are few-shot learners. arXiv [cs.CL]. 10.48550/arXiv.2005.14165

Cabrera-Diego, Luis Adrián & Akshita Gheewala. 2023. Jus Mundi at SemEval-2023 Task 6: Using a frustratingly easy domain adaption for a legal named entity recognition system. Em *17th International Workshop on Semantic Evaluation (SemEval)*, 1783–1790. 10.18653/v1/2023.semeval-1.247

Chalkidis, Ilias, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras & Ion Androutsopoulos. 2020. LEGAL-BERT: The Muppets straight out of law school. Em *Findings of the Association for Computational Linguistics*, 2898–2904. 10.18653/v1/2020.findings-emnlp.261

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer & Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. Em *58th Meeting of the Association for Computational Linguistics*, 8440–8451. 10.18653/v1/2020.acl-main.747

Correia, Fernando A., Alexandre A. A. Almeida, José Luiz Nunes, Kaline G. Santos, Ivar A. Hartmann, Felipe A. Silva & Hélio Lopes. 2022. Fine-grained legal entity annotation: A case study on the Brazilian Supreme Court. *Information Processing & Management* 59(1). 102794. 10.1016/j.ipm.2021.102794

Costa, Rosimeire, Hidelberg Oliveira Albuquerque, Gabriel Silvestre, Nádia Félix F. Silva, Ellen Souza, Douglas Vitório, Augusto Nunes, Felipe Siqueira, João Pedro Tarrega, João Vitor Beinotti, Márcio de Souza Dias, Fabíola S. F. Pereira, Matheus Silva, Miguel Gardini, Vinicius Silva, André C. P. L. F. de Carvalho & Adriano L. I. Oliveira. 2022. Expanding UlyssesNER-Br named entity recognition corpus with informal user-generated text. Em *EPIA Conference on Artificial Intelligence*, 767–779. 10.1007/978-3-031-16474-3_62

Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 4171–4186. 10.18653/v1/N19-1423

Douka, Stella, Hadi Abdine, Michalis Vazirgiannis, Rajaa El Hamdani & David Restrepo Amariles. 2021. JuriBERT: A masked-language model adaptation for French legal text. Em *Natural Legal Language Processing Workshop (NLLP)*, 95–101. 10.18653/v1/2021.nllp-1.9

Firdaus Solihin, Rizal Fathoni Aji, Indra Budi & Edmon Makarim. 2021. Advancement of information extraction use in legal documents. *International Review of Law, Computers & Technology* 35(3). 322–351. 10.1080/13600869.2021.1964225

Garcia, Eduardo A. S., Nadia F. F. Silva, Felipe Siqueira, Hidelberg O. Albuquerque, Juliana R. S. Gomes, Ellen Souza & Eliomar A. Lima. 2024. RoBERTaLexPT: A legal RoBERTa model pretrained with deduplication for Portuguese. Em *16th International Conference on Computational Processing of Portuguese (PROPOR)*, 374–383.

Gururangan, Suchin, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey & Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. Em *58th Meeting of the Association for Computational Linguistics (ACL)*, 8342–8360. 10.18653/v1/2020.acl-main.740

Gutiérrez-Fandiño, Asier, Jordi Armengol-Estapé, Aitor Gonzalez-Agirre & Marta

Villegas. 2021. Spanish legalese language model and corpora. arXiv [cs.CL/cs.AI]. 10.48550/arXiv.2110.12201

Har-Peled, Sariel, Piotr Indyk & Rajeev Motwani. 2012. Approximate nearest neighbors: Towards removing the curse of dimensionality. *Theory of Computing* 8(1). 321–350. 10.4086/toc.2012.v008a014

Jaccard, Paul. 1912. The distribution of the flora in the alpine zone. *New phytologist* 11(2). 37–50. 10.1111/j.1469-8137.1912.tb05611.x

Kaplan, Jared, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu & Dario Amodei. 2020. Scaling laws for neural language models. arXiv [cs.LG/stat.ML]. 10.48550/arXiv.2001.08361

Kapoor, Arnav, Mudit Dhawan, Anmol Goel, Arjun T H, Akshala Bhatnagar, Vibhu Agrawal, Amul Agrawal, Arnab Bhattacharya, Ponnurangam Kumaraguru & Ashutosh Modi. 2022. HLDC: Hindi legal documents corpus. Em *Findings of the Association for Computational Linguistics*, 3521–3536. 10.18653/v1/2022.findings-acl.278

Lee, Jinhyuk, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So & Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36(4). 1234–1240. 10.1093/bioinformatics/btz682

Lee, Katherine, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch & Nicholas Carlini. 2022. Deduplicating training data makes language models better. Em *60th Meeting of the Association for Computational Linguistics (ACL)*, 8424–8445. 10.18653/v1/2022.acl-long.577

Licari, Daniele & Giovanni Comandè. 2022. ITALIAN-LEGAL-BERT: A pre-trained transformer language model for italian law. Em *23rd International Conference on Knowledge Engineering and Knowledge Management*,

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer & Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. arXiv [cs.CL]. 10.48550/arXiv.1907.11692

Luz de Araujo, Pedro Henrique, Teófilo E. De Campos, Renato R. R. De Oliveira, Matheus Stauffer, Samuel Couto & Paulo Bermejo. 2018. LeNER-Br: A dataset for named entity recognition in Brazilian legal text. Em *Computational Processing of the Portuguese Language (PROPOR)*, 313–323. 10.1007/978-3-319-99722-3_32

Masala, Mihai, Radu Cristian Alexandru Iacob, Ana Sabina Uban, Marina Cidota, Horia Velicu, Traian Rebedea & Marius Popescu. 2021. jurBERT: A Romanian BERT model for legal judgement prediction. Em *Natural Legal Language Processing Workshop (NLLP)*, 86–94. 10.18653/v1/2021.nllp-1.8

Menezes-Neto, Elias Jacob de & Marco Bruno Miranda Clementino. 2022. Using deep learning to predict outcomes of legal appeals better than human experts: A study with data from Brazilian federal courts. *PLOS ONE* 17(7). e0272287. 10.1371/journal.pone.0272287

Niklaus, Joel, Veton Matoshi, Matthias Stürmer, Ilias Chalkidis & Daniel Ho. 2024. MultiLegalPile: A 689GB multilingual legal corpus. Em *62nd Meeting of the Association for Computational Linguistics (ACL)*, 15077–15094. 10.18653/v1/2024.acl-long.805

Ott, Myle, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier & Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. Em *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 48–53. 10.18653/v1/N19-4009

Paul, Shounak, Arpan Mandal, Pawan Goyal & Saptarshi Ghosh. 2023. Pre-trained language models for the legal domain: a case study on indian law. Em *19th International Conference on Artificial Intelligence and Law (ICAIL)*, 187–196. 10.1145/3594536.3595165

Polo, Felipe, Gabriel Mendonça, Kauê Parreira, Lucka Gianvechio, Peterson Cordeiro, Jonathan Ferreira, Leticia Lima, Antônio Maia & Renato Vicente. 2021. LegalNLP - natural language processing methods for the Brazilian legal language. Em *18th Encontro Nacional de Inteligência Artificial e Computacional*, 763–774. 10.5753/eniac.2021.18301

Real, Livy, Erick Fonseca & Hugo Gonçalo Oliveira. 2020. The ASSIN 2 shared task: A quick overview. Em *Computational Processing of the Portuguese Language (PROPOR)*, 406–412. 10.1007/978-3-030-41505-1_39

Rodrigues, João, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso & Tomás Osório. 2023. Advancing neural encoding of portuguese with transformer Albertina PT-*. Em *EPIA Conference on Artificial Intelligence*, 441–453. 10.1007/978-3-031-49008-8_35

Santos, Diana, Nuno Seco, Nuno Cardoso & Rui Vilela. 2006. HAREM: An advanced NER evaluation contest for Portuguese. Em *International Conference on Language Resources and Evaluation (LREC)*, 1986–1991.

Siqueira, Felipe A., Douglas Vitório, Ellen Souza, José A. P. Santos, Hidelberg O. Albuquerque, Márcio S. Dias, Nádia F. F. Silva, André C. P. L. F. de Carvalho, Adriano L. I. Oliveira & Carmelo Bastos-Filho. 2024. Ulysses Tesemõ: a new large corpus for brazilian legal and governmental domain. *Language Resources and Evaluation* 10.1007/s10579-024-09762-8

Souza, Fábio, Rodrigo Nogueira & Roberto Lotufo. 2020. BERTimbau: Pretrained BERT models for Brazilian Portuguese. Em Ricardo Cerri & Ronaldo C. Prati (eds.), *Brazilian Conference on Intelligent Systems (BRACIS)*, 403–417. 10.1007/978-3-030-61377-8_28

Tirumala, Kushal, Daniel Simig, Armen Aghajanyan & Ari Morcos. 2023. D4: Improving LLM pretraining via document de-duplication and diversification. Em *Advances in Neural Information Processing Systems*, vol. 36, 53983–53995.

Viegas, Charles F. O., Bruno Catais Costa & Renato Porfirio Ishii. 2023. JurisBERT: Transformer-based model for embedding legal texts. Em *International Conference on Computational Science and Its Applications (ICCSA)*, 349–365. 10.1007/978-3-031-36805-9_24

Wagner Filho, Jorge A., Rodrigo Wilkens, Marco Idiart & Aline Villavicencio. 2018. The brWaC corpus: A new open resource for Brazilian Portuguese. Em *11th International Conference on Language Resources and Evaluation (LREC)*, 4339–4344.

Willian Sousa, Antonio & Marcos Fabro. 2019. Iudicium Textum Dataset: Uma base de textos jurídicos para NLP. Em *14th Simpósio Brasileiro de Banco de Dados*, 1–11.

Xiao, Chaojun, Xueyu Hu, Zhiyuan Liu, Cunchao Tu & Maosong Sun. 2021. Lawformer: A pre-trained language model for Chinese legal long documents. *AI Open* 2. 79–84. 10.1016/j.aiopen.2021.06.003

Xue, Linting, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua & Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. Em *North American Chapter of the Association for Computational Linguistics (NAACL)*, 483–498. 10.18653/v1/2021.naacl-main.41

Zhong, Haoxi, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu & Maosong Sun. 2020. How does NLP benefit legal system: A summary of legal artificial intelligence. Em *58th Meeting of the Association for Computational Linguistics (ACL)*, 5218–5230. 10.18653/v1/2020.acl-main.466

Modelo	***Dataset***	***Learning Rate***	**Batch Size**	**Passos**	**Épocas**	**F1-*Score* (Validação)**
RoBERTaTimbau$_{base}$	LeNER-Br	2.5e-5	16	3000	6.1	89.68
RoBERTaTimbau$_{base}$	UlyssesNER-Br T.	5.e-5	16	1750	12.2	89.73
RoBERTaTimbau$_{base}$	UlyssesNER-Br C.	5.e-5	16	3750	26.2	88.79
RoBERTaTimbau$_{base}$	FGV-STF Amplo	5.e-5	32	3000	15.0	77.93
RoBERTaTimbau$_{base}$	RRI	5.e-5	16	1500	2.9	83.17
RoBERTaCrawlPT$_{base}$	LeNER-Br	7.5e-6	32	3500	14.2	89.76
RoBERTaCrawlPT$_{base}$	UlyssesNER-Br T.	5.e-5	16	750	5.2	88.74
RoBERTaCrawlPT$_{base}$	UlyssesNER-Br C.	2.5e-5	16	2000	14.0	88.27
RoBERTaCrawlPT$_{base}$	FGV-STF Amplo	2.5e-5	16	5000	13.8	76.55
RoBERTaCrawlPT$_{base}$	RRI	5.e-5	32	3000	11.6	82.82
RoBERTaLegalPT$_{base}$	LeNER-Br	5.e-5	32	2000	8.1	90.62
RoBERTaLegalPT$_{base}$	UlyssesNER-Br T.	5.e-5	16	750	5.2	88.62
RoBERTaLegalPT$_{base}$	UlyssesNER-Br C.	5.e-5	16	2000	14.0	88.36
RoBERTaLegalPT$_{base}$	FGV-STF Amplo	5.e-5	16	2500	7.6	75.98
RoBERTaLegalPT$_{base}$	RRI	5.e-5	16	1000	1.9	82.94
RoBERTaLexPT$_{base}$	LeNER-Br	1.e-5	32	3750	15.2	89.83
RoBERTaLexPT$_{base}$	UlyssesNER-Br T.	5.e-5	32	2250	31.2	90.71
RoBERTaLexPT$_{base}$	UlyssesNER-Br C.	5.e-5	32	1750	24.3	89.12
RoBERTaLexPT$_{base}$	FGV-STF Amplo	2.5e-5	16	4000	11.9	75.28
RoBERTaLexPT$_{base}$	RRI	5.e-5	16	4000	7.7	84.29
RoBERTaLexPT+$_{base}$	LeNER-Br	5.e-5	16	2750	5.6	90.39
RoBERTaLexPT+$_{base}$	UlyssesNER-Br T.	5.e-5	16	1750	12.2	90.07
RoBERTaLexPT+$_{base}$	UlyssesNER-Br C.	5.e-5	16	2250	15.7	89.36
RoBERTaLexPT+$_{base}$	FGV-STF Amplo	5.e-5	16	2000	6.0	75.11
RoBERTaLexPT+$_{base}$	RRI	5.e-5	16	1000	1.9	84.33
RoBERTaLexPT$_{large}$	LeNER-Br	1.e-5	16	3750	7.6	89.64
RoBERTaLexPT$_{large}$	UlyssesNER-Br T.	2.5e-5	16	2000	14.0	89.81
RoBERTaLexPT$_{large}$	UlyssesNER-Br C.	5.e-5	32	3250	45.1	88.83
RoBERTaLexPT$_{large}$	FGV-STF Amplo	5.e-5	32	1500	8.9	75.87
RoBERTaLexPT$_{large}$	RRI	7.5e-6	32	500	1.9	83.77
RoBERTaLexPT+$_{large}$	LeNER-Br	7.5e-6	16	1750	3.6	90.37
RoBERTaLexPT+$_{large}$	UlyssesNER-Br T.	2.5e-5	16	1000	7.0	89.62
RoBERTaLexPT+$_{large}$	UlyssesNER-Br C.	1.e-5	16	3500	24.5	90.12
RoBERTaLexPT+$_{large}$	FGV-STF Amplo	5.e-5	16	4000	11.9	75.51
RoBERTaLexPT+$_{large}$	RRI	2.5e-5	32	500	1.9	84.78

Tabela 12: Melhores configurações de hiperparâmetros para modelos na fase de ajuste fino do PortuLex. O F1-*score* corresponde ao Macro F1-*Score* no conjunto de validação de cada dataset.

Modelo	***Dataset***	***Learning Rate***	***Batch Size***	**Passos**	**Épocas**	**Pontuação (Validação)**
BERTimbau$_{base}$	ASSIN2 RTE	1.e-5	16	3250	8.0	97.20
BERTimbau$_{base}$	ASSIN2 STS	2.5e-5	16	5750	14.1	96.87
BERTimbau$_{base}$	HAREM Total	5.e-5	32	750	46.9	74.21
BERTimbau$_{base}$	HAREM Seletivo	5.e-5	32	500	31.2	86.05
BERTimbau$_{large}$	ASSIN2 RTE	2.5e-5	32	750	3.7	96.40
BERTimbau$_{large}$	ASSIN2 STS	5.e-5	32	5000	24.5	97.38
BERTimbau$_{large}$	HAREM Total	5.e-5	16	1000	32.3	75.75
BERTimbau$_{large}$	HAREM Seletivo	5.e-5	16	750	24.2	87.82
Albertina-PT-BR$_{base}$	ASSIN2 RTE	7.5e-6	32	2500	12.2	96.40
Albertina-PT-BR$_{base}$	ASSIN2 STS	2.5e-5	16	7250	17.8	97.32
Albertina-PT-BR$_{base}$	HAREM Total	2.5e-5	16	1250	27.2	74.19
Albertina-PT-BR$_{base}$	HAREM Seletivo	5.e-5	16	1500	32.6	85.30
Albertina-PT-BR$_{xlarge}$	ASSIN2 RTE	7.5e-6	16	1500	3.7	97.40
Albertina-PT-BR$_{xlarge}$	ASSIN2 STS	1.e-5	32	5250	25.7	97.88
Albertina-PT-BR$_{xlarge}$	HAREM Total	1.e-5	32	1000	47.6	80.08
Albertina-PT-BR$_{xlarge}$	HAREM Seletivo	2.5e-5	32	1000	47.6	91.33
RoBERTaTimbau$_{base}$	ASSIN2 RTE	2.5e-5	32	2000	9.8	96.40
RoBERTaTimbau$_{base}$	ASSIN2 STS	2.5e-5	32	5750	28.2	96.99
RoBERTaTimbau$_{base}$	HAREM Total	2.5e-5	16	1250	43.1	74.34
RoBERTaTimbau$_{base}$	HAREM Seletivo	5.e-5	32	750	50.0	83.15
RoBERTaCrawPT$_{base}$	ASSIN2 RTE	2.5e-5	32	3250	15.9	96.60
RoBERTaCrawPT$_{base}$	ASSIN2 STS	5.e-5	32	4500	22.1	97.03
RoBERTaCrawPT$_{base}$	HAREM Total	2.5e-5	16	1000	34.5	77.67
RoBERTaCrawPT$_{base}$	HAREM Seletivo	5.e-5	16	750	25.9	86.30
RoBERTaLegalPT$_{base}$	ASSIN2 RTE	7.5e-6	16	5500	13.5	95.60
RoBERTaLegalPT$_{base}$	ASSIN2 STS	5.e-5	32	4000	19.6	96.96
RoBERTaLegalPT$_{base}$	HAREM Total	5.e-5	16	750	24.2	74.90
RoBERTaLegalPT$_{base}$	HAREM Seletivo	5.e-5	16	1000	32.3	83.91
RoBERTaLexPT+$_{base}$	ASSIN2 RTE	1.e-5	32	1750	8.6	97.00
RoBERTaLexPT+$_{base}$	ASSIN2 STS	5.e-5	32	4000	19.6	96.90
RoBERTaLexPT+$_{base}$	HAREM Total	5.e-5	16	1250	43.1	79.02
RoBERTaLexPT+$_{base}$	HAREM Seletivo	5.e-5	16	1250	43.1	91.12
RoBERTaLexPT+$_{large}$	ASSIN2 RTE	7.5e-6	32	1750	8.6	97.00
RoBERTaLexPT+$_{large}$	ASSIN2 STS	1.e-5	32	3250	15.9	97.57
RoBERTaLexPT+$_{large}$	HAREM Total	1.e-5	16	1250	43.1	78.87
RoBERTaLexPT+$_{large}$	HAREM Seletivo	1.e-5	16	500	17.2	87.28

Tabela 13: Melhores hiperparâmetros para os modelos na fase de ajuste fino dos *benchmarks* ASSIN2 e HAREM. A pontuação corresponde à métrica principal: coeficiente de correlação de Pearson para ASSIN2 STS e Macro F1-Score para os demais datasets.

Proposta recebida em Outubro 2024 e aceite para publicação em Dezembro 2024.

Rotulação e Caracterização de Conteúdo Tóxico de Comunidades do Reddit no Brasil

Toxic Content Detection in Online Social Networks: A New Dataset from Brazilian Reddit Communities

Luiz Henrique Quevedo Lima
Universidade Federal de Minas Gerais

Ana Clara Souza Pagano
Universidade Federal de Minas Gerais

Adriana Silvina Pagano
Universidade Federal de Minas Gerais

Ana Paula Couto da Silva
Universidade Federal de Minas Gerais

Resumo

A ausência de dados de qualidade em idiomas com baixa disponibilidade de recursos, como o Português brasileiro, é um desafio significativo para a moderação automatizada de conteúdo online. Nos últimos anos, a proliferação de interações sociais online e o crescimento de conteúdo gerado por usuários trouxeram à tona a questão crescente da linguagem tóxica. Embora modelos automáticos de aprendizado de máquina tenham sido eficazes na moderação do vasto volume de dados nas redes sociais, ferramentas eficientes para esses idiomas ainda são escassas. Neste trabalho, tratamos essa lacuna criando um conjunto de dados de alta qualidade, coletado de algumas das comunidades brasileiras mais populares da plataforma Reddit. Para isso, rotulamos manualmente um conjunto de 2.500 comentários extraídos das comunidades com maior engajamento e número de inscritos. Realizamos uma análise exploratória para encontrar achados valiosos sobre a linguagem de conteúdo *tóxico* e *não-tóxico*. Nossos resultados mostram um nível moderado de concordância entre os anotadores, validando a relevância desse conjunto de dados para diversas tarefas de aprendizado de máquina. Esta pesquisa busca contribuir para a criação de um ambiente online mais seguro para os usuários que participam de discussões virtuais, além de abrir caminho para o desenvolvimento de ferramentas de moderação automática mais eficazes baseadas em aprendizado de máquina.

Palavras chave

toxicidade; português; conjunto de dados

Abstract

The proliferation of online social interactions in recent years, with the consequent growth in user-generated content, has brought the escalating issue of toxic language. While automatic machine learning models have been effective in moderating the vast amount of data on online social networks, low-resource languages, such as Brazilian Portuguese, still lack efficient automated moderation tools. We address this gap by creating a novel dataset collected from some of the most popular Brazilian Reddit communities. To that end, we manually labeled a sample dataset of 2,500 comments extracted from the most engaging communities. We conducted an in-depth exploratory analysis to gain valuable insights into the language of *toxic* and *non-toxic* content. Our results show a high level of agreement among annotators, attesting to the suitability of this dataset for various downstream machine learning tasks. This research offers a significant contribution to the creation of a safer online environment for users engaging in discussions in Portuguese and paves the way for more effective automatic moderation tools using machine learning.

Keywords

toxicity; portuguese; dataset

1. Introdução

O aumento no número de plataformas de redes sociais estimula cada vez mais as interações de usuários nestas mídias. De acordo com Statista (2022), o número total de usuários das diferentes redes sociais já alcançou os 4 bilhões de pessoas. Esse número sinaliza o nível de importância e onipresença dessas plataformas na sociedade e seu impacto, nem sempre benéfico, na vida das pessoas. De acordo com Vogels (2021), um estudo realizado em 2020 com adultos norte-americanos descobriu que cerca de 41% dos entrevistados haviam experimentado alguma forma de assédio online. Além disso, comentários abusivos em discussões propagam a toxicidade, levando à radicalização das discussões (Salehabadi et al., 2022).

DOI: 10.21814/lm.16.2.459
This work is Licensed under a Creative Commons Attribution 4.0 License

As consequências dessas interações transcendem o mundo virtual, afetando seriamente a vida dos usuários no mundo real. De acordo com Vogels (2021), 18% dos usuários que participaram de uma enquete sofreram algum tipo de abuso, considerado grave, fora do ambiente online, incluindo ameaças físicas e perseguição.

A moderação manual do conteúdo gerado por usuários tem sido considerada a principal abordagem para atenuar o impacto negativo das interações tóxicas. No entanto, a escala e a velocidade com que o conteúdo é gerado tornam a moderação manual impraticável, o que leva à necessidade de soluções automatizadas (Gillespie, 2020). Os modelos de aprendizado de máquina surgiram como uma alternativa promissora para automatizar a moderação de conteúdo criado online. Esses modelos podem identificar conteúdo potencialmente prejudicial, permitindo que as plataformas tomem medidas proativas, como banir usuários e remover conteúdo prejudicial. Embora os modelos de aprendizado de máquina tenham se mostrado eficazes em vários idiomas (Google, 2022b), seu desempenho em línguas que possuem menos recursos, como o português brasileiro, ainda é insuficiente.

Face a esses desafios, neste trabalho apresentamos um novo conjunto de dados destinado à detecção de toxicidade em português brasileiro. Os textos anotados foram extraídos de uma das redes sociais online mais numerosas —Reddit—, que possui cerca de 1,5 bilhões de usuários registrados e 430 milhões de usuários ativos (Wise, 2023). Reddit é uma comunidade que permite que os usuários interajam por meio de postagens anônimas e comentários. Os usuários se agrupam em comunidades (subreddits) que escolhem por serem mais mais alinhadas com seus tópicos de interesse. Visando propor novos modelos de detecção de toxicidade e aprimorar os já existentes para características específicas da língua portuguesa, realizamos a coleta e a anotação de dados. Nosso conjunto de dados é adaptado para dados de redes sociais online, especificamente com dados do Reddit, uma plataforma que possui características bem distintas das comumente consideradas, como X e Instagram. Além disso, apesar da existência de mecanismos de moderação, muitas publicações ainda contêm conteúdo tóxico, o que reforça a relevância de nossa contribuição no preenchimento da lacuna de dados e modelos disponíveis treinados para o português nesse domínio.

Este artigo está organizado da seguinte forma. Na Seção 2, apresentamos uma síntese da literatura disponível sobre detecção de toxicidade em português. Em seguida, na Seção 3, apresentamos nossa metodologia de coleta e anotação de dados, bem como as técnicas utilizadas para caracterizar a linguagem das postagens coletadas. Na Seção 4, apresentamos os resultados do nosso estudo. Primeiramente, fazemos uma descrição geral do conjunto de dados, para depois relatar nosso experimento no qual comparamos a anotação humana dos dados com a anotação automática por meio da API Perspective e de dois grandes modelos de linguagem (LLMs). Em seguida, caracterizamos a linguagem usada nos comentários *tóxicos* e *não-tóxicos*. A Seção encerra-se com a discussão dos principais achados e seu potencial impacto, sobretudo no que diz respeito ao uso deste conjunto de dados para ajustar os modelos de classificação de toxicidade existentes. Apresentamos também as limitações do nosso estudo. Por último, a Seção 5 apresenta as conclusões do nosso estudo, no escopo de iniciativas que visam aprimorar a moderação automática de conteúdo em um ambiente online em constante crescimento. Ao abordar as deficiências dos recursos existentes, nosso objetivo é contribuir com os esforços para tornar as redes sociais online mais seguras e inclusivas para todos. Para permitir a reprodutibilidade e promover estudos de acompanhamento, publicamos o conjunto de dados anotados para acesso público.[1]

2. Artigos relacionados

Os estudos sobre a detecção automática de comentários tóxicos em idiomas como o português brasileiro são escassos, bem como os conjuntos de dados anotados manualmente e de livre acesso para uso público e estudos de acompanhamento.

de Pelle & Moreira (2017) disponibilizam um conjunto de dados com 1.250 comentários, extraídos de sessões de comentários do site `https://g1.globo.com` e anotados com as categorias ofensivo e não ofensivo, sendo 32,5% do total rotulado como ofensivo. A classe de comentários ofensivos foi subdividida em *racismo, sexismo, homofobia, xenofobia, intolerância religiosa e xingamentos*. Os xingamentos, incluindo linguagem vulgar, foram a categoria mais frequente de comentários ofensivos, presentes em quase 70% dos comentários considerados ofensivos.

Fortuna et al. (2019) descrevem um conjunto de dados com 5.668 tweets, rotulados com um esquema de anotação hierárquica por anotadores com diferentes níveis de expertise. Os anotadores

[1] O conjunto de dados está disponível em `https://github.com/luizhenriqueds/reddit-br-toxicity-dataset/`.

Trabalho	Conjunto de dados	# exemplos	# classes	Anotadores	Especialistas	Tarefa
de Pelle & Moreira (2017)	Comentários de notícias	1,250	2	*Crowdsourcing*	✗	Detecção de linguagem ofensiva
Fortuna et al. (2019)	Twitter	5,668	2+	Especialistas e não especialistas	✓	Classificação binária; Comentários são classificados em 81 diferentes subclasses por especialistas
Leite et al. (2020)	Twitter	21,000	6	42 estudantes voluntários	✗	Multiclasse: insulto, misogenia, racismo, etc.
Vargas et al. (2022)	Instagram	7,000	2+	Não informado	✓	Detecção de linguagem ofensiva e nível de ofensa (se aplicável)
Trajano et al. (2023)	YouTube e Twitter	7,943	5	5 anotadores contratados	✗	Detecção de discurso de ódio; detecção de toxicidade

Tabela 1: Resumo das características dos trabalhos relacionados à construção de conjuntos de dados em Português do Brasil. Trabalhos que estão marcados com *2+* classes indicam estudos de anotação hierárquica, onde uma tarefa inicial de classificação binária (*ofensivo* vs *não-ofensivo*, *tóxico* vs *não-tóxico*, por exemplo) é decomposta em esquemas de anotação mais detalhados e de grão fino.

não especialistas anotaram os tweets com rótulos binários (*ódio vs. não-ódio*).Em seguida, os anotadores mais experientes utilizaram um esquema hierárquico para classificar com maior granularidade os tweets, fazendo uso de 81 categorias relativas ao discurso do ódio.

Leite et al. (2020) apresentam ToLD-Br: um conjunto de dados para a classificação de comentários tóxicos no Twitter em português brasileiro. Um total de 21 mil tweets foi anotado manualmente com sete categorias: *não tóxico, LGBTQ+fobia, obsceno, insulto, racismo, misoginia e xenofobia*. Cada tweet foi rotulado por três anotadores voluntários de uma universidade brasileira. Por meio de uma análise ampla e abrangente, eles comprovam a necessidade de se desenvolver conjuntos de dados específicos por língua para estudos de classificação automática de comentários tóxicos.

O desempenho da API Perspective no português brasileiro é avaliado por Kobellarz & Silva (2022). Os comentários de dois sites brasileiros de mídia de notícias foram traduzidos para o inglês e sua toxicidade foi rotulada pela Perspective tanto em sua versão original em português e sua tradução para o inglês. Os resultados da rotulação automática foram comparados com a anotação humana. A comparação evidenciou melhor desempenho da API em textos em seu idioma original.

O corpus HateBR é apresentado por Vargas et al. (2022). O corpus abrange 7.000 comentários de contas de Instagram de políticos brasileiros, anotados manualmente por especialistas, com uma alta concordância entre os anotadores. Os documentos possuem três tipos de anotação: uma classificação binária (comentários ofensivos versus não ofensivos), nível de ofensividade (altamente, moderadamente e levemente ofensivo) e nove categorias de discurso de ódio (*xenofobia, racismo, homofobia, sexismo, intolerância religiosa, partidarismo, apologia à ditadura, antissemitismo e gordofobia*).

Trajano et al. (2023) apresentam OLID-BR, um conjunto de dados de alta qualidade para detecção de linguagem ofensiva. O conjunto de dados contém 6.354 (extensível a 13.538) comentários rotulados usando um esquema de anotação de três camadas compatível com conjuntos de dados em outros idiomas, o que permite o treinamento de modelos multilíngues.

O trabalho de Oliveira et al. (2023) comparou o desempenho do ChatGPT usando a abordagem *zero-shot* na tarefa de detecção de discurso de ódio em tweets em Português. Os conjuntos de dados utilizados foram o ToLD.BR e o HLPHSP. Os modelos comparados incluem: BERTimbau, ChatGPT-3.5 com variação de *prompts*, DistilBERT e um modelo linear. Os resultados mostraram que o ChatGPT pode superar modelos treinados especificamente para a tarefa, alcançando *score* F1 de 73% e 74% nos conjuntos de dados, respectivamente. Além disso, os resultados indicaram que o desempenho é muito sensível à engenharia de prompts.

A Tabela 1 sumariza as principais características dos trabalhos relacionados apresentados nesta seção. Nosso trabalho contribui para os estudos sobre caracterização de conteúdo tóxico ao investigar comentários em português brasileiro publicados em redes sociais online. Este é o primeiro estudo que enfoca a criação e caracterização de um corpus do Reddit em português brasileiro, anotado manualmente quanto à toxicidade.

3. Metodologia

Nesta seção, primeiro descrevemos nossa metodologia de coleta do corpus. Em seguida, descrevemos o processo para rotular manualmente uma amostra de comentários como *tóxico* e *não-tóxico*. Por fim, apresentamos os métodos usados para analisar a linguagem dos comentários rotulados.

Subreddit	Inscritos	Postagens	Commentários
r/brasil	1,516,433	110,829	2,382,928
r/desabafos	490,049	115,876	1,487,076
r/futebol	369,925	35,826	1,272,009
r/saopaulo	358,681	7,308	88,894
r/eu_nvr	308,064	12,631	221,348
r/botecodoreddit	270,451	7,059	62,999
r/conversas	247,545	21,967	355,761
r/investimentos	232,485	9,756	156,695
r/tiodopave	219,926	2,371	12,106
r/brasilivre	210,582	67,301	1,308,441
	Total	390,924	7,348,257

Tabela 2: Subreddits selecionados, número de assinantes, postagens e comentários do ano 2022.

3.1. Coleta de dados

Reddit é uma rede social online multilíngue fundada em 2005 e organizada em subcomunidades por áreas de interesse (subreddits). Nosso conjunto de dados consiste em atividades de usuários (postagens e comentários) entre janeiro e dezembro de 2022 nos 10 principais subreddits brasileiros com o maior número de assinantes[2] e com uma vida útil de pelo menos cinco anos, o que atesta sua importância nessa rede social online.

A Tabela 2 apresenta os subreddits selecionados e algumas estatísticas descritivas. Coletamos um total de 7.348.257 comentários e 390.924 postagens por meio do Pushshift, uma API de terceiros que agrega comentários e publicações do Reddit (Baumgartner et al., 2020). Em nossa análise, utilizaremos a denominação *comentários* para referir-nos tanto aos comentários quanto às postagens feitas pelos usuários.

Nosso conjunto de dados inclui comentários apenas em português, sendo excluídos comentários de comunidades que permitem discussões em vários idiomas. Aproximadamente 600 mil comentários, nos quais o texto foi substituído por *deletado* ou *removido*, foram excluídos da análise, bem como comentários contendo apenas emojis ou símbolos, URLs, caracteres não alfanuméricos e reações de texto apenas de risada.[3] Por fim, também excluímos comentários gerados por contas de automoderadores e bots que detectamos em nossos dados. Esses filtros reduziram nosso corpus para aproximadamente 6,6 milhões de comentários. A Tabela 3 apresenta algumas estatísticas para os subreddits analisados após a aplicação dos filtros.

Subreddit	Postagens	Comentários
r/brasil	115,876	2,136,866
r/desabafos	115,876	1,211,643
r/futebol	35,826	1,214,412
r/saopaulo	7,308	81,969
r/eu_nvr	12,631	188,620
r/botecodoreddit	7,059	57,298
r/conversas	21,967	326,061
r/investimentos	9,756	141,823
r/tiodopave	2,371	11,584
r/brasilivre	67,301	1,219,265
Total	390,924	6,589,541

Tabela 3: Subreddits selecionados e total de postagens e comentários (2022) após a filtragem.

3.2. Processo de anotação

Primeiramente, extraímos uma amostra de 2.500 comentários do nosso corpus após aplicados os filtros, usando um processo de amostragem estratificada que preservou a distribuição original do número total de comentários por mês em cada subreddit. Essa amostra de comentários foi dividida em 5 Lotes de 500 textos cada. Em seguida, recrutamos 12 alunos de graduação e pós-graduação dos cursos de Ciência da Computação e Estudos da Linguagem de uma universidade brasileira para participarem como anotadores. Os alunos foram divididos em 4 grupos e receberam instruções sobre como rotular cada comentário do Reddit com uma de quatro categorias: *Tóxico*, *Não-tóxico*, *Não sei* ou *Informações insuficientes para rotular o conteúdo*.[4] Para fins de anotação, consideramos linguagem tóxica todo comentário *rude, desrespeitoso ou irracional com*

[2] De acordo com o ranking de Almerekhi et al. (2019).

[3] Em português, textos de risadas são representados pela sequência de caracteres "*kkkkk*"

[4] *Informações insuficientes para rotular o conteúdo* foi uma categoria incluída para futura investigação em um estudo sobre propagação de toxicidade no Reddit.

grande probabilidade de fazer com que alguém saia de uma discussão, conforme definido pela API Perspective. Cada grupo recebeu um Lote e cada comentário foi rotulado por três anotadores independentes. Um dos grupos recebeu um Lote adicional de comentários, dada a alta qualidade da anotação realizada por eles, conforme será discutido na Seção 4.1.

Para a rotulação final do conjunto de dados, a cada comentário do Reddit é atribuída uma das categorias quando há um consenso majoritário entre os anotadores. Aplicamos três métricas para medir a concordância entre os avaliadores: a estatística Kappa de Fleiss, o Alfa de Krippendorf e a Concordância Observada.

3.3. Caracterização da linguagem

Para investigar padrões na linguagem dos comentários de conteúdo tóxico em língua portuguesa, adotamos os seguintes procedimentos na análise do conjunto de dados anotados manualmente.

3.3.1. Identificação automática de comentários tóxicos

Para medir a correlação entre a identificação automática e manual de conteúdo tóxico na amostra de comentários do Reddit, selecionamos três diferentes modelos de aprendizado de máquina. O primeiro deles é o modelo implementado pela API Perspective (Google, 2022b). A API Perspective é um conjunto de classificadores de toxicidade prontos para uso do Google Jigsaw, que foi amplamente usado em pesquisas anteriores (Almerekhi et al., 2020; Salehabadi et al., 2022; Zannettou et al., 2020; ElSherief et al., 2018). A API recebe um comentário como entrada e retorna uma pontuação de 0 a 1 para vários classificadores (por exemplo, palavrões, ameaças, ataques à identidade, toxicidade geral). Com relação ao idioma português, Google (2022a) relatam uma área sob a curva ROC (AUC) de 0,89 para a tarefa de classificação do modelo. Para enriquecer a análise de concordância entre classificação automática e classificação manual, selecionamos dois modelos representantes dos grandes modelos de linguagem (em inglês, *Large Language Models* ou LLMs): o chat GPT-3.5 (Brown et al., 2020) e o Sabia-2 (Almeida et al., 2024). Estes modelos têm sido cada vez mais utilizados para realizar diferentes tarefas de processamento de linguagem (Kukreja et al., 2024).

3.3.2. Análise de classe de palavras

Para caracterizar a linguagem dos comentários *tóxicos* e *não-tóxicos*, exploramos a frequência das palavras utilizadas e sua classe (Part-of-Speech). Para etiquetar a classe de palavra (Petrov et al., 2011), usamos um modelo pré-treinado (spaCy, 2022) e um banco de árvores em português (treebank) anotado com sintaxe de dependência de acordo com as diretrizes das Dependências Universais (Rademaker et al., 2017). O modelo selecionado possui uma precisão de mais de 97%.

Calculamos a frequência das etiquetas de classe de palavra nos comentários *tóxicos* e *não-tóxicos* a fim de descobrir se essa poderia ser uma característica distintiva dos dois tipos de comentários.

3.3.3. Razão type-token e extensão dos comentários

Para computar a razão type-token, dividimos o número total de palavras não repetidas ("types") pelo número total de palavras "tokens"). Também comparamos o tamanho dos comentários *tóxicos* e *não-tóxicos*. Diferentemente de outras redes sociais online, no Reddit não há restrições para o tamanho do comentário; portanto, essa medida permite calcular a probabilidade de os usuários publicarem um texto curto ou longo na plataforma.

3.3.4. Análise de tópicos

Para extrair os tópicos dos comentários em que os anotadores mais concordam ou discordam, executamos o modelo BERTopic (Grootendorst, 2022), que se baseia em uma representação vetorial para agrupar documentos semelhantes.

3.3.5. Análise de n-gramas

N-gramas são sequências de n tokens consecutivos de um determinado conjunto de dados, usados para representação da ocorrência de termos. Formalmente, definimos um n-grama como uma sequência de palavras $w_1, w_2, \ldots, w_n$, onde cada w_i representa um token no corpus. Neste artigo, apresentamos resultados para unigramas ($n = 1$) e para bigramas ($n = 2$).

3.3.6. Grafos de coocorrência

Uma análise complementar de um corpus pode ser feita por meio da observação de redes de coocorrência. Para obter essa rede, contamos o

número de coocorrências de cada par de palavras encontradas nos comentários analisados. Em seguida, definimos um grafo onde os vértices representam as palavras e as arestas indicam se existe coocorrência nos comentários coletados. Para a construção do grafo, consideramos os pares de palavras que ocorreram pelo menos duas vezes. Os nós do grafo são termos mencionados pelo menos 50 vezes nos comentários analisados.

3.3.7. Reconhecimento de entidades nomeadas

Investigamos entidades nomeadas nos comentários do Reddit com base em um modelo pré-treinado do Spacy para reconhecimento de entidades nomeadas (NER). O modelo usado foi treinado para o português brasileiro usando o conjunto de dados WikiNER (Nothman et al., 2013) e classifica as entidades em três categorias predefinidas: PESSOA, LOCALIZAÇÃO e ORGANIZAÇÃO. As entidades não definidas são classificadas como DIVERSAS.

4. Resultados

Nesta seção, apresentamos os principais resultados obtidos com a avaliação e a caracterização do conjunto de dados anotado manualmente.

4.1. Concordância entre anotadores

Primeiramente, calculamos a concordância global entre os anotadores nos comentários do Reddit rotulados manualmente, cujos resultados são mostrados na Tabela 4.

Métrica	Global	Rótulos binários (*Não-tóxico* ou *Tóxico*)
Kappa de Fleiss	0,31	0,46
Alfa de Krippendorff	0,35	0,46
Concordância Observada	0,64	0,80

Tabela 4: Concordância entre anotadores.

Como esperado, a métrica de *concordância observada* obteve os valores mais altos, pois essa medida não leva em conta a possibilidade de a concordância ocorrer por acaso. Houve total concordância e total discordância em 1.594 e 107 comentários, respectivamente. Um exemplo de concordância total sobre um comentário considerado *tóxico* é: *"Como assim? Eu nem sou o OP. Só tô dizendo que ele é retardado de seguir a medicina de gado"*. Por outro lado, um exemplo de discordância total é um comentário polêmico como: *"[..] é o lugar do Brasil que mais tem neonazi mesmo ué"*, o que aponta para o alto nível de subjetividade da tarefa de classificação.

Com relação às métricas *Kappa de Fleiss* e *Alfa de Krippendorff*, os valores indicam concordância de razoável a moderada no pior dos casos. Por fim, a toxicidade geral classificada pelos anotadores foi de 11,28%, com 88,7% de comentários *não-tóxicos*, o que é consistente com a natureza desbalanceada do problema.

Em seguida, calculamos a concordância entre os anotadores de cada grupo, denominados A, B, C e D, para os Lotes de comentários, numerados como 1, 2, 3, 4 e 5. Os Lotes 3 e 5 foram anotados pelo grupo C, enquanto os Lotes 1, 2 e 4 foram anotados pelos grupos A, B e D, respectivamente. O Lote 5 foi rotulado em uma segunda rodada de anotação pelo grupo C, selecionado para isso por ser o grupo que obteve o maior valor de *Kappa de Fleiss* e *Alfa de Krippendorff* para a concordância entre anotadores na primeira rodada. A Tabela 5 mostra os resultados. Com exceção do Grupo D, que obteve de leve a nenhuma concordância, os grupos A, B e C obtiveram uma concordância de razoável a moderada.

Em seguida, examinamos a rotulagem feita por cada anotador, cujos resultados são mostrados na Tabela 6. O grupo A rotulou a menor porcentagem de comentários como *tóxico*. Já o Grupo B apresenta a maior variabilidade na rotulagem de conteúdo tóxico, sendo o anotador 2 o que rotulou mais de 21% dos comentários como *tóxicos*. Assim como o Grupo B, o Grupo D atingiu um nível não negligenciável de incerteza na tarefa de classificação, sendo que o anotador 2 tendeu a ser mais tolerante com o conteúdo *tóxico* em potencial. Para fins de ilustração, o comentário *"Vamos fingir que não é (você) que posta que quer morrer por ser depressivo. Pick me boy"* foi classificado como *tóxico* pelos anotadores 1 e 3 e como *não-tóxico* pelo anotador 2. Os anotadores do Grupo C, que receberam os Lotes 3 e 5, são os que apresentam o menor grau de incerteza.

Em geral, nossos resultados corroboram o alto nível de subjetividade inerente à tarefa de classificar conteúdo como *tóxico* ou *não-tóxico*. Isso está de acordo com os resultados da literatura sobre como a percepção do grau de severidade do conteúdo nocivo é afetada por valores individuais e culturais (Jiang et al., 2021).

Métrica	Lote 1	Lote 2	Lote 3	Lote 4	Lote 5
Kappa de Fleiss	0,46	0,33	0,51	0,17	0,54
Alfa de Krippendorff	0,46	0,33	0,51	0,17	0,54
Concordância observada	0,87	0,81	0,76	0,78	0,77

Tabela 5: Métricas de concordância entre anotadores por Lote de anotação.

	Lote 1 (Grupo A)			Lote 2 (Grupo B)			Lote 3 (Grupo C)		
	Anot. 1	Anot. 2	Anot. 3	Anot. 1	Anot. 2	Anot. 3	Anot. 1	Anot. 2	Anot. 3
Não-tóxico	84,60%	88,96%	90,60%	83,17%	69,48%	74,95%	75,90%	68,01%	78,51%
Tóxico	9,40%	9,84%	7,40%	7,82%	21,29%	4,81%	19,28%	21,73%	17,87%
Não sei dizer	0,60%	1,00%	0,00%	3,81%	3,82%	2,81%	4,02%	7,65%	3,01%
Informação insuficiente	5,40%	0,20%	2,00%	5,21%	5,42%	17,43%	0,80%	2,62%	0,60%

	Lote 4 (Grupo D)			Lote 5 (Grupo C)		
	Anot. 1	Anot. 2	Anot. 3	Anot. 1	Anot. 2	Anot. 3
Não-tóxico	72,80%	93,59%	69,14%	84,51%	68,60%	75,20%
Tóxico	11,60%	5,21%	9,02%	14,49%	25,00%	19,72%
Não sei dizer	4,20%	0,80%	6,41%	1,01%	4,20%	4,67%
Informação insuficiente	11,40%	0,40%	15,43%	0,00%	2,20%	0,41%

Tabela 6: Distribuição dos rótulos de anotação para cada grupo de anotadores.

4.2. Comparação entre a rotulagem manual e as rotulagens automáticas

4.2.1. API Perspective

Comparamos nossa anotação manual de dados com a realizada pela API Perspective. Consideramos tóxicos os comentários aos quais a API Perspective atribuiu uma pontuação de **toxicidade grave** acima de 0,7. Essa decisão prioriza um bom equilíbrio entre precisão e revocação, pois nossa intenção é compreender melhor os principais motivos de concordância e discordância na classificação de conteúdo *tóxico* e *não-tóxico*. Um valor limite de 0,9 resulta na seleção de apenas 3% dos comentários tóxicos para comparação. Em contrapartida, um valor de 0,7 retorna aproximadamente 10% dos comentários como tóxicos, uma porcentagem semelhante àquela rotulada por nossos anotadores.

Limiar	Precisão	Revocação	F1	#Tóxico
0,5	0,65	0,69	0,67	92
0,6	0,69	0,62	0,65	78
0,7	0,8	0,41	0,55	45
0,8	0,81	0,4	0,54	43
0,9	1,00	0,15	0,26	13

Tabela 7: Desempenho da API Perspective no conjunto de dados de teste com distintos limiares de escore de toxicidade.

Porcentagem de toxicidade Primeiro, analisamos a porcentagem de comentários anotados como *tóxicos* por nossos voluntários e a porcentagem rotulada pela API Perspective. O Grupo A (Lote 1) anotou menos comentários tóxicos do que a API Perspective, enquanto um anotador do Grupo B (Lote 2) classificou uma porcentagem muito maior de comentários como *tóxicos*. O Grupo C (Lotes 3 e 5) evidencia uma anotação consistente de maior número de comentários tóxicos do que API Perspective. O Grupo D (Lote 4), apesar de mostrar grande discordância entre anotadores, também evidenciou um número menor de comentários tóxicos do que a API. A Tabela 7 mostra o desempenho da API Perspective em uma amostra de teste rotulada pelo Grupo C. O objetivo da análise não é comparar diretamente a concordância entre anotadores humanos e a API Perspective, mas, sim, avaliar a qualidade das previsões da API em diferentes limiares em um conjunto de teste com curadoria. Os resultados indicam uma clara compensação (trade-off) no desempenho entre precisão e revocação. Na prática, ao escolher um limiar de precisão alto, estamos compensando uma grande parte do desempenho de revocação. Portanto, o modelo treinado do Perspective tem uma grande margem de melhoria para textos em português do Brasil, considerando os limiares selecionados. Combinando as métricas de revocação e precisão, obtemos um escore F1 máximo de 0,67.

Correlação entre as anotações tóxicas Calculamos como os rótulos manuais e os rótulos da API Perspective se correlacionam entre si. A correlação geral de Benesty et al. (2009) na amostra de teste é de 0,51, comparando o rótulo do voto majoritário para cada comentário. Também calculamos a correlação entre grupos de anotado-

res e as previsões da API Perspective. Os anotadores dos Lotes 1, 2 e 3 apresentaram uma correlação moderada consistente com a API, enquanto os anotadores do Lote 4 apresentaram uma correlação fraca. Por fim, os anotadores do Lote 5 apresentaram uma correlação consistente e forte com a API.

4.2.2. Grandes modelos de linguagem

Para comparar os resultados de grandes modelos de linguagens (LLMs) com as anotações manuais, classificamos o mesmo conjunto de dados previamente anotado usando os modelos GPT-3.5 e Sabia-2. O primeiro modelo é treinado com dados de diversos idiomas (inclusive o Português do Brasil), enquanto o segundo é uma versão ajustada (*fine-tuned*) para dados em Português. Para ambos os modelos, definimos o *prompt* que indica, de forma bem estruturada e clara, as principais instruções para a realização da tarefa de classificação de toxicidade. A Figura 1 mostra o *prompt* utilizado para a tarefa de classificação. Para reproduzir as mesmas condições do experimento com anotadores, essas instruções foram baseadas nas mesmas instruções fornecidas aos anotadores para fazer a anotação manual do conjunto de dados. Por fim, para ambos os modelos, consideramos o tamanho máximo de tokens de entrada de cada modelo igual a 500 e temperatura[5] igual a zero.

> Você é um assistente e deve atuar como moderador de conteúdo de redes sociais, responsável por classificar comentários do Reddit em português brasileiro (PT-BR) como *tóxicos* ou *não-tóxicos*.
>
> Siga as instruções abaixo para realizar a tarefa:
>
> **1.** Um comentário é considerado *tóxico* se for um texto ofensivo ou rude em relação a outra pessoa em publicações de redes sociais;
>
> **2.** Você receberá o texto de um comentário, e sua tarefa é classificar a toxicidade do texto fornecido. Use apenas essas informações para fazer a previsão;
>
> **3.** Para cada comentário, classifique-o em uma das seguintes categorias: *Não-tóxico* ou *Tóxico*.

Figura 1: Instruções fornecidas ao assistente para classificação de comentários no Reddit usando a abordagem *zero-shot*.

[5]Temperatura é um parâmetro que permite modificar a saída de um modelo de linguagem, tornando-as mais previsível ou criativa.

Porcentagem de toxicidade Ao analisar a proporção de comentários classificados como *tóxico* pelos modelos, observamos que o modelo Sabiá-2 se aproxima da distribuição do percentual de toxicidade por Lotes observada para a Perspective API, variando entre 9,7% para o Lote 4 e 12,09% para o Lote 5, com média 11,45%. Em contrapartida, as classificações de toxicidade do ChatGPT foram maiores em média (28,40%), variando entre 29,2% para o Lote 2 e 31,4% para o Lote 3. Esse resultado evidencia uma maior sensibilidade do ChatGPT ao teor de toxicidade dos comentários analisados no período, enquanto o Sabiá-2 é menos influenciado por nuances na semântica das discussões, uma vez que o percentual de classificação se aproxima da distribuição das anotações manuais. Como exemplo, o seguinte comentário foi classificado como *tóxico* pelo ChatGPT e *não-tóxico* pelos anotadores *"Isso é bobagem, país nenhum tá livre de corrupção, só algum paiseco inexpressivo talvez, tipo a nova Zelândia. O que o povo quer de verdade é uma vida de qualidade e com segurança, e isso nem Bolsonaro nem Lula chegam perto de promover"*, enquanto o mesmo acontece com o Sabiá-2 classificando o seguinte comentário *"Fala que roubado é mais gostoso"*.

Para analisar a sensibilidade dos modelos a um comentário que foi classificado como *tóxico* de forma unânime, submetemos o seguinte comentário *"Posso fazer a piada do bebe morto?"* aos LLMs e tanto o ChatGPT quanto o Sabia-2 classificaram o teor deste comentário como *tóxico*, em contraste com a Perspective API que o classificou como *não-tóxico*.

Os resultados da comparação entre a anotação manual e a anotação automática revelam diferenças relevantes no comportamento dos modelos Sabiá-2 e ChatGPT em relação à classificação de toxicidade. A Tabela 8 mostra que o modelo Sabiá-2 apresentou um equilíbrio entre falsos positivos (128) e falsos negativos (126), demonstrando uma abordagem mais balanceada. Em contrapartida, o ChatGPT teve um número consideravelmente maior de falsos positivos (454), mas um número significativamente menor de falsos negativos (43), indicando que, embora mais propenso a classificar erroneamente comentários *não-tóxicos* como *tóxicos*, o ChatGPT foi mais eficaz na identificação de comentários realmente *tóxicos*. Esses resultados corroboram com a análise anterior de que o Sabia-2 é menos conservador, enquanto o ChatGPT é mais sensível à presença de toxicidade.

		Lt 1	Lt 2	Lt 3	Lt 4	Lt 5	Total
Sabia-2	**FP**	33	30	18	36	11	**128**
	FN	16	11	46	11	42	**126**
ChatGPT	**FP**	106	88	82	113	65	**454**
	FN	6	4	15	5	13	**43**
	Total	**161**	**133**	**161**	**165**	**131**	

Tabela 8: Falso positivos (FP) e falso negativos (FN) por grupo.

Essas características, embora diferentes, podem ser aplicadas a depender do cenário em questão. Por exemplo, se o interesse for minimizar a presença de falsos negativos, pode ser interessante empregar um modelo que seja mais sensível à presença de toxicidade e, por sua vez, consiga capturar esses casos com mais precisão. Entretanto, pensando na quantidade massiva de comentários em redes sociais online, uma abordagem híbrida que alcance o maior equilíbrio entre os dois tipos de erro pode ser mais eficiente.

Correlação entre as anotações tóxicas Ao comparar a classificação dos anotadores com os modelos, observa-se uma correlação moderada entre as classes geradas pelos modelos e anotadores. A correlação entre os anotadores e o Sabiá-2 é de 0,43, enquanto o ChatGPT teve uma correlação um pouco menor (0,41). Ao analisar a correlação por Lotes do conjunto de dados, tivemos a menor correlação para o Lote 4 (Sabia-2: 0,22, ChatGPT: 0,59) e a maior correlação para o Lote 5 (Sabia-2: 0,21, ChatGPT: 0,58), reforçando a alta concordância do Grupo C com modelos de classificação automática. Para explorarmos a concordância entre as classes, calculamos o Kappa de Cohen entre as anotações e as classificações dos modelos. A Tabela 9 mostra o Kappa score para cada grupo. Observa-se, novamente, menor concordância entre o Grupo D e maior concordância com os Grupo C.

	Lt 1	Lt 2	Lt 3	Lt 4	Lt 5
Sabiá-2	0,3964	0,3583	0,4396	0,2072	0,5779
ChatGPT-3.5	0,2628	0,2418	0,4511	0,1277	0,5616

Tabela 9: *Kappa score* entre anotações e modelos de linguagem (PT-BR) instruídos para executar a tarefa de classificação de toxicidade.

Por fim, comparamos o resultado dos modelos de classificação automática. O modelo pré-treinado da Perspective API nos permite controlar a flexibilidade das classificações, uma vez que um *score* de toxicidade é calculado. No limiar de 0,7, o modelo alcançou precisão de 0,8 e *recall* de 0,41. Nesse limiar, o modelo é capaz de classificar comentários como *tóxico* com mais precisão, ao custo de deixar de recuperar algumas instâncias. Para os modelos de linguagem, não temos acesso ao *score* com a confiança do modelo. Entretanto, através de engenharia de *prompt*, instruímos os LLMs a favorecer a precisão (ou seja, favorecer casos em que o modelo tem alta confiança de que um comentário é realmente *tóxico*). O Sabia-2 apresentou equilíbrio entre as métricas de precisão e revocação (precisão: 0,71; revocação: 0,72), enquanto o ChatGPT favoreceu muito mais a métrica de precisão (precisão: 0,8; revocação: 0,64). Todos os modelos de classificação automática foram influenciados pela presença de gíriais locais e palavrões, como discutido anteriormente.

4.3. Caracterização da linguagem do conteúdo tóxico e não tóxico

Comparamos os padrões na linguagem do conteúdo *tóxico* e *não-tóxico* para entender melhor como os falantes de português utilizam a linguagem para gerar conteúdo tóxico.

4.3.1. Análise da extensão dos comentários e razão type-token

Com relação à extensão dos comentários, o número médio de tokens e o intervalo de confiança de 95% para comentários *não-tóxicos* é 26,34 [24,68, 28,19]. Para os comentários *tóxicos* a média é de 35,54 [29,41, 42,87]. Portanto, os comentários *tóxicos* são, em média, mais longos do que os *não-tóxicos* (p-valor $< 0,05$). A distribuição do comprimento nos comentários *tóxicos* tem um intervalo maior, o que pode indicar diferenças dentro dos próprios subreddits.

A média da razão type-token e o intervalo de confiança para os comentários *não-tóxicos* é 0,78 [0,78, 0,79], enquanto para os comentários *tóxicos* é 0,83 [0,82, 0,84]. Os resultados apontam para significância estatística, tendo os comentários *tóxicos* mais diversidade lexical. Isso pode variar entre os subreddits, pois algumas das comunidades são mais propensas a ter postagens mais verbosas.

4.3.2. Análise de etiquetas de POS

A diversidade de etiquetas de classe de palavra (POS) para comentários *não-tóxicos* tem uma média de 0,51 [0,50, 0,52], enquanto que para textos *tóxicos* rotulados a média é de 0,46 [0,43, 0,48]. Embora os comentários *tóxicos* sejam mais longos, eles geralmente são menos diversificados em termos de etiquetas de POS.

Para investigar melhor as etiquetas de POS, comparamos a distribuição de algumas etiquetas específicas. Primeiro, comparamos os adjetivos (ADJ) com uma média de 1,68 [1,55, 1,81] para comentários *não-tóxicos* e 2,14 [1,71, 2,66] para comentários *tóxicos*. Como os intervalos de confiança se sobrepõem entre as classes, realizamos um teste estatístico Mann-Whitney para comparar as diferenças nas distribuições. O uso da etiqueta ADJ é estatisticamente diferente entre as classes, com um valor de $p < 0{,}01$.

Da mesma forma, realizamos o mesmo teste para a etiqueta NOUN, da classe de palavra substantivo. O uso médio em comentários *não-tóxicos* é de 5,43 [5,07, 5,83], enquanto nos comentários *tóxicos* a média é de 7,44 [6,15, 8,94]. Essa diferença é novamente validada pelo teste Mann-Whitney, com um valor de $p < 0{,}01$.

Uma análise da distribuição de etiquetas de POS nos comentários é essencial para entender as características do texto gerado pelos usuários do Reddit nas maiores comunidades brasileiras. Para isso, usamos o marcador de POS pré-treinado do Spacy para o português brasileiro. Cada token em uma frase foi classificado com uma das etiquetas de POS existentes. À lista de etiquetas de POS, foram adicionadas outras classes específicas para o problema de classificação de etiquetas, como SYM, SPACE e X para denotar "símbolos", "espaço em branco" e "outros", respectivamente, com a observação de que, como esse é um modelo de aprendizado de máquina treinado em corpora pertencentes a outros domínios, a classificação pode resultar em falsos positivos.

As duas etiquetas de POS mais comuns para comentários *tóxicos* e *não-tóxicos* são NOUN (substantivo) e VERB (verbo). Os comentários *não-tóxicos* usam mais etiquetas PROPN (nome próprio), enquanto uma alta porcentagem de tokens de comentários *tóxicos* foi etiquetada como PUNCT (sinal de pontuação). Além disso, os comentários *tóxicos* usam mais interjeições, etiquetadas como INTJ. Também comparamos as distribuições de etiquetas de POS de ambas as classes por meio de um teste de qui-quadrado. Os resultados indicam que a diferença observada entre a distribuição das etiquetas de POS é significativa (p-valor $< 0{,}05$).

Para uma análise mais detalhada das diferenças no uso de palavras nos comentários *tóxicos* e *não-tóxicos*, extraímos as palavras mais frequentes por classe de comentário para as etiquetas de POS mais frequentes, ou seja, as etiquetas ADJ, NOUN e PROPN. Os resultados são mostrados na Tabela 10. Uma descoberta relevante é o termo *mulher* em comentários *tóxicos*. De fato, realizamos um teste de qui-quadrado para comparar a associação desse termo com comentários *tóxicos* e *não-tóxicos* Os resultados indicam uma associação positiva para alguns dos subreddits brasileiros (como o r/desabafos) com valor de $p < 0{,}05$. Esse resultado pode sugerir a presença de comportamento misógino associado a alguns tópicos e comunidades nas redes sociais. Exemplos de comentários direcionados a mulheres nas discussões das comunidades podem ser encontrados na (Tabela 14).

4.3.3. Análise de tópicos

Investigamos os comentários sobre os quais os anotadores discordaram totalmente, especialmente em relação aos principais tópicos extraídos com o modelo BERTopic. Eles dizem respeito a discussões relacionadas a grupos específicos (mulheres, homens) e abrangem vários temas, incluindo finanças, guerra, governo e relacionamentos (Tabela 11). As palavras no tópico 0 (*feedback, remoção*) revelam que alguns comentários foram moderados anteriormente pela DMCA (Digital Millennium Copyright Act, (US Congress, 1998)).

É interessante notar que os principais tópicos dos comentários sobre os quais os anotadores concordaram totalmente também discutem os mesmos temas (Tabela 12). Entretanto, os descritores de tópicos incluem mais termos ofensivos (como palavrões) e ideologicamente carregados.

Com relação aos tópicos extraídos dos comentários em que os três anotadores concordaram no rótulo *tóxico* e que a API Perspective classificou como *não-tóxico* (Tabela 13), os principais têm a ver com política, liberdade, discriminação e grupos-alvo. Os resultados indicam que a API é menos sensível ao contexto para essa tarefa específica do português brasileiro. Por exemplo, o comentário a seguir foi rotulado como *tóxico* por todos os três anotadores, mas foi classificado como *não-tóxico* pelo modelo de aprendizado de máquina *"Posso fazer a piada do bebe morto?"*. Ao comparar com LLMs, vemos um comportamento similar. Por exemplo, o seguinte comentário foi classificado como *tóxico* pelos anotadores e *não-tóxico* pelo ChatGPT: *"Não, acho que deveriam fazer rinha de fake news, f*da-se, no fim, 'todo mundo' recebe e sabe discernir e prefere a informação verdadeira."*, enquanto o comentário *"Nojento! E bizarro o quão frequente isso é. Só demonstra que a nossa sociedade é machista e doente."* foi classificado como *tóxico* pelo processo de anotação e *não-tóxico* pelo Sabia-2.

	ADJ	NOUN	PROPN
Não-tóxico	bom, melhor, mesmo, grande, mesma, pior, fácil, diferente	cara, gente, pessoas, coisa, tempo, anos, vida, mundo, dinheiro	Brasil, Lula, Bolsonaro, OP, Deus, Flamengo, Landau, Ciro, PT, STF
Tóxico	melhor, mesmo, pobre, ruim, primeiro, forte, diferente, social, capaz, política, rico	pessoas, cara, mundo, mulher, c*, casa, homem, m**da, pai	Lula, Bolsonaro, Brasil, OP, Ciro, Ucrânia, Flamengo, FDP, Liberdade, Rússia, Paris

Tabela 10: Palavras mais frequentes por etiqueta de POS e classe de comentário.

Tópico	Descritores
0	video, mulher, opinião, homem, dinheiro, beleza, burro , padrão, feedback, removal
1	guerra, liberdade, post, motivo, país, massacres, massa, atrocidades, históricas, democracia, governo, xenofóbico
2	m**da, sexo, maluco, apoiadores, preocupado, machão, malditos, insegurança, op (original poster)

Tabela 11: Tópicos e palavras-chave relevantes em comentários em que os três anotadores discordaram (desacordo total).

A Tabela 14 mostra exemplos de comentários direcionados à *mulher* diretamente em comentários *tóxicos*. Alguns dos comentários causaram total desacordo entre os anotadores. Por exemplo, o comentário *"Pelo direito de bater na própria mulher! Uow"* foi rotulado como 'Não sei', 'Tóxico' e 'Não tóxico'. Uma hipótese é que o texto seja interpretado como um comentário sarcástico ou irônico. Um experimento adicional com mais contexto (como fornecer a sequência da conversa ao anotador) pode mitigar o desacordo nesses casos. Comentários que exigem detalhes contextuais são difíceis de rotular, mesmo para anotadores humanos, e ainda mais para modelos de aprendizado de máquina treinados em corpora que não se assemelham a interações em redes sociais. Na verdade, para esse comentário específico, a API do Perspective previu como *não-tóxico* com as configurações de parâmetros padrão. Em relação aos LLMs, o ChatGPT classificou corretamente os dois primeiros exemplos da tabela como *tóxico* e o Sabiá-2 classificou os três comentários como *não-tóxico*.

4.3.4. Análise de n-gramas

A Figura 2 apresenta os termos (unigramas) mais frequentes considerando todos os comentários do conjunto de dados. Termos como *pessoas, melhor, gente, ano, mundo, Brasil* aparecem em grande destaque, refletindo o foco nas discussões sobre pessoas, eventos globais e relacionados ao país. Além desses termos, encontramos termos como *vida, problema, caso, amigo*, que sugerem discussões sobre questões pessoais, desafios e relacionamentos, assuntos que são frequentemente discutidos em algumas das principais comunidades, como r/desabafos e r/conversas. Essas comunidades incentivam a troca de ideias acerca de assuntos do dia-a-dia e discussões sobre relacionamentos. Por fim, a presença de termos como *Lula, Bolsonaro* mostra que a política está entre os assuntos mais discutidos, refletindo a polarização e o engajamento nas questões de política interna, principalmente considerando que os dados foram coletados em 2022, ano de eleição presidencial no Brasil.

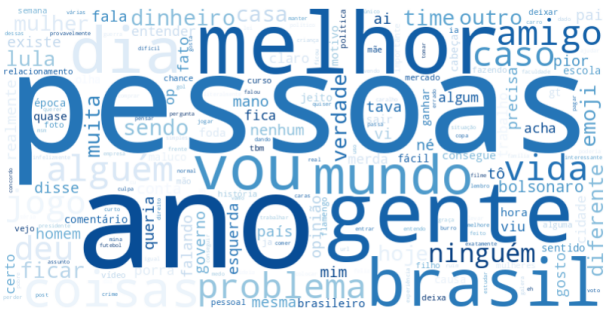


Figura 2: Unigramas mais frequentes encontrados nos comentários das principais comunidades brasileiras no Reddit.

Já a Figura 3 mostra os unigramas mais frequentes divididos por classe de comentários. Nos comentários *tóxicos*, palavras como *Lula, Bolsonaro, m*a, p*a, burro* se destacam, muitas vezes associadas a discussões sobre política e insultos. É possível notar que a polarização política é predominante nas discussões —uma vez que os dois principais candidatos à pre-

Tópico	Descritores
0	burro, homem, p**ra, c*, mulher, mercado, gente, anos, país, b**ta, criança, ódio, sentido
1	guerra, bolsonaro, ucrânia, realidade, putin, intervenção, pobre, nuclear, bandido, vergonha, russia
2	ideologia, liberdade, política, mundo, cancelamento, expressão, op (original poster), preconceito, oprimidos, vagabundo, família

Tabela 12: Tópicos e palavras-chave relevantes em comentários em que todos os três anotadores classificaram como *tóxico*.

Tópico	Descritores
0	ideologia, política, liberdade, mundo, pessoas, expressão, mulheres, preconceito, bolsonaro, esquerdistas, apolíticos, piada, realidade, oprimidos, opiniões

Tabela 13: Palavras-chaves extraídas de comentários em que todos os três anotadores classificaram como *tóxicos* e que a API do Perspective previu como *não-tóxicos* (falsos negativos).

sidência em 2022 são mencionados— e insultos direcionados a outros usuários da plataforma. Além disso, é possível observar a presença de termos que indicam discussões mais ideológicas ou culturais, como *governo, ideologia, política*. Por fim, o termo *guerra* também é mencionado com frequência nos comentários classificados como *tóxicos*, o que possivelmente foi causado pelo estopim da guerra Rússia-Ucrânia no período de coleta dos dados, revelando o uso difundido dessa rede social para discutir assuntos inerentes da globalização mundial.

Para uma análise simplificada de contexto, a Tabela 15 apresenta os bigramas mais recorrentes nas comunidades analisadas neste artigo. Conforme mencionado anteriormente, durante o período de coleta dos dados houve a eleição presidencial no Brasil, o que explica a alta frequência de termos como *votar lula*, *liberdade expressão* em comentários classificados como tóxicos, revelando o cenário de polarização interna (*"Eu deixo claro minha opinião: sou contra liberdade de expressão(tem gente que não sabe o que diz) e sou contra a democracia (tem gente que não sabe votar)... kkkkk mas respeito ambas as coisas por que sei que a b*s*a deste mundo não é perfeito"*). Além disso, observa-se a presença da expressão *tio sam*, usada como referência aos Estados Unidos. Por fim, há uma forte ocorrência de palavrões e insultos, reforçando a associação deste tipo de linguagem com toxicidade percebida por humanos, como, por exemplo, o comentário *"fit cultural é de cair o c* da bunda. Desculpa para qualquer coisa p*p"*.

No caso dos comentários *não-tóxicos*, a análise dos bigramas revela uma variedade de temas presentes nas principais comunidades brasileiras do Reddit. Nos subreddits r/desabafos e r/brasil, os termos mais frequentes estão relacionados a reflexões pessoais e discussões construtivas sobre temas sociais e políticos. Bigramas como *muita gente*, *ler escrever* e *ficar sozinha* refletem interações mais introspectivas e emocionais, típicas de conversas sobre experiências individuais e desabafos, por exemplo: *"Não, nem um pouco. Você não chorar não significa que não amava a pessoa ou não a considerava. Tem muita gente que não consegue chorar com essas coisas; e isso é completamente normal"*. Já em r/brasil, aparecem tópicos mais amplos como *rock rio*, *redes sociais*, e figuras políticas como *ciro gomes*.

No subreddit r/brasilivre, observa-se uma concentração de termos como *processo eleitoral* e *opinião pública*, sugerindo debates politizados e possíveis temas controversos. Além disso, é possível observar uma grande prevalência de comentários removidos nessa comunidade, indicados pelos termos *view, removal* e *removal, request*. Em r/futebol, temos bigramas como *time jogando* e *flamengo culpa*, esperados dada a natureza do subreddit. No entanto, ainda há a presença de termos relacionados às eleições e políticas externas, como, por exemplo *ministério público* e *guerra*.

Por fim, no subreddit r/investimentos, os bigramas revelam uma ênfase em tópicos educacionais e profissionais, com termos como *lógica matemática*, *atividades extracurriculares*, e *curso voltado*, indicando discussões focadas no desenvolvimento pessoal e financeiro. Na comunidade

Comentário	Texto
1	*"A minoria quer realmente ser independente - mas como o universo do /r/brasil é majoritariamente progressista, não irão concordar - as demais estão entre o "mulher tem que ser mulher" e aquelas que usam o discurso de independência, mas acham que quem tem que pagar as coisas é o homem."*
2	*"O mundo é assim, do mesmo jeito que você não quer uma mulher feia, uma mulher não vai querer alguém feio ou sem status, não cai nesse papo de que aparência não importa que em rede social só tem alienado, veja você mesmo pesquisas relacionadas ao assunto ou se tiver coragem crie um perfil com a foto de alguém bonito e veja como as pessoas te tratam diferente."*
3	*"[..] Mas o homem casa com quem ele quiser. A mulher casa com quem ela consegue."*

Tabela 14: Exemplos de comentários mencionando a palavra "mulher" em comentários *tóxicos*.

Figura 3: Termos frequentes em comentários classificados como *tóxico* (esquerda) e comentários classificados como *não-tóxicos* (direita).

r/conversas, também temos a prevalência da moderação de conteúdo, onde comentários são frequentemente removidos pela moderação por provavelmente infringirem direitos autorais ou regras da comunidade.

Em resumo, é possível observar que termos políticos dominam os assuntos discutidos e estão presentes em praticamente todas as comunidades. Além disso, em comunidades consideradas mais liberais, temos maior incidência de moderação ou conteúdo removido por direitos autorais. Consideramos comunidades liberais, comunidades que discutem majoritariamente assuntos considerados polêmicos, o que pode influenciar na tendência dos usuários a tecer comentários tóxicos e/ou ataques direcionados. Nesse tipo de discussão, pode ter uma influência maior do uso de termos pejorativos ou palavrões e isso influenciar na decisão de modelos automáticos, por exemplo. Além disso, algumas comunidades - como a r/brasilivre - incentivam abertamente o debate livre entre os participantes. Segundo a sua definição no Reddit: *O subreddit brasileiro mais livre do reddit. Nosso objetivo é proporcionar a todos um espaço aberto e livre para debates.*

4.3.5. Grafo de coocorrência

O grafo de coocorrência extraído dos comentários possui o total de 9373 nós e 41844 arestas, com grau médio das arestas igual a 6,11 para termos associados com comentários *tóxicos* e 10,33 para termos associados com comentários *não-tóxico*. A Figura 4 apresenta a métrica de centralidade de nó baseada no grau do nó do grafo. Esta métrica é definida por $C_D(v) = \deg(v)$ e mede o número de conexões que um nó (termo) tem com outros nós no grafo. Em uma matriz de coocorrência, os termos influentes são termos que coocorrem com muitos outros termos no conjunto. O grau de cada nó a partir da matriz de adjacências é definido pela soma $\deg(v) = \sum_{u \in V} A_{vu}$.

Para os comentários *tóxicos*, *mundo, pessoas, gente* aparecem com alta centralidade, refletindo seu papel central nas discussões polarizadas, frequentemente ligadas a temas políticos, como demonstrado pela presença de *esquerda, Bolsonaro, Lula*. Esses termos indicam que debates políticos intensos geram alta toxicidade, especialmente em torno de figuras públicas e ideologias.

Para os comentários classificados com *não-tóxicos*, como *coisas, dia, pessoas, hoje*, possuem alta centralidade de grau e representam discussões mais cotidianas e neutras, voltadas a in-

	Tóxico	**Não-tóxico**
r/desabafos	{preferir, ficar}, {compra, carro}, {olhar, ser}, {tomar, c*}, {compra, carro}, {liberdade, expressão}, {existem, pessoas}, {ato, libidinoso} , {pessoa, tratar}, {querer, alguém}, {maltratar, animal}	{muita, gente}, {ler, escrever}, {ficar, sozinha}, {material, genético}, {possa, ajudar}, {mano, foda}, {carne, magra}, {diferente, alguém}, {auto, estima}, {hoje, dia}, {queria, alguém}, {tô, falando}
r/brasil	{liberdade, expressão}, {pé, chão}, {sonho, americano}, {virar, prof}, {sonhar, trabalho}, {hipócrita, dissimular}, {opinião, diferente}, {mulheres, pretos}	{muita, gente}, {hoje, dia}, {dia, seguinte}, {rock, rio}, {álcool, isopropílico}, {durar, anos}, {redes, sociais}, {segue, vida}, {gente, falando}, {ciro, gomes}, {vale, pena}, {nota, fiscal}
r/brasilivre	{existir, pessoa}, {ir, votar}, {sociedade, viver}, {tio, sam}, {mínimo, respeito}, {chupa, c*}, {pessoa, preferir}	{muita, gente}, {view, removal}, {removal, request}, {request, video}, {processo, eleitoral}, {juros, dividas}, {velho, testamento}, {esquerda, autoritária}, {pior, esquerda}, {opinião, pública}, {vamos, ignorar}, {carga, viral}
r/futebol	{tomar, c*}, {entender, futebol}, {comparação, sentido}, {ficar, calado}, {mano, neymar}, {andar, campo}	{time, jogando}, {deus, existe}, {funções, diferentes}, {flamengo, culpa}, (graças, deus}, {bola, parada}, {coréia, sul}, {fisica, nuclear}, {jogo, grêmio}, {viu, guerra}, {viu, surgimento}, {viu, queda}
r/investimentos	{operar, vender}, {ironia, lula}, {lula, vencedor}, {precificado, fato}, {verdade, operar}, {rir, toa}	{lógica, matemática}, {engenharias, vida}, {curso, voltado}, {atividades, extracurriculares}, {boas, indicações}, {manter, dinheiro}, {opções, risco}, {celebração, imigração}, {imigração, japonesa}, {clt, real}, {real, empresa}, {empresa, brasileira}
r/conversas	{mau, aluno}, {ir, fingir}, {postar, morte}, {morrer, depressivo}, {pick, boy}, {pick, m*rda}	{envie, mensagem}, {post, comentário}, {url, remoção}, {remoção, envie}, {mensagem, apelar}, {apelar, post}, {comentário, removido}, {removido, diretório}, {diretório, envie}, {pede, sair}

Tabela 15: Bigramas mais frequentes entre as principais comunidades brasileiras no Reddit. Os conjuntos de termos estão segmentados entre os termos associados com comentários classificados como *tóxicos* e *não-tóxicos*.

terações descritivas e informais. Termos como *melhor, vida, Brasil* também se destacam, sugerindo que a *não-toxicidade* está associada a conversas mais propositivas e positivas. Assim, grafos de coocorrência podem auxiliar na moderação automatizada de conteúdo tóxico, pois revelam, com uma modelagem simples, palavras que frenquentemente são utilizadas em conjunto nestes tipos de discussão.

4.3.6. Reconhecimento de entidades nomeadas (NER)

A Tabela 16 apresenta os resultados da análise de entidades nomeadas realizada em nosso conjunto de dados. A entidade nomeada mais comum em ambas as classes é PESSOA, representando mais de 31% de todos os tokens classificados em comentários tóxicos. A segunda entidade mencionada com mais frequência, LOCALIZAÇÃO, é igualmente predominante em ambas as classes. Embora tanto os comentários *tóxicos* quanto os *não-tóxicos* mencionem essas entidades, seu uso é diferente. Realizamos um teste de qui-quadrado para comparar a distribuição de etiquetas de POS para comentários em que pelo menos uma entidade nomeada é mencionada. O resultado indica uma diferença significativa na distribuição das etiquetas de POS (p-valor $< 0{,}01$). Os comentários *tóxicos* por exemplo, usam mais tokens etiquetados como VERB e NOUN. O comentário a seguir exemplifica entidades nomeadas sendo mencionadas na discussão dos usuários: *"Mais sério que esse tweet só a guerra na Ucrânia"*.

Conteúdo	**PER**	**ORG**	**LOC**	**MISC**
Não-tóxico	28,49%	20,26%	26,35%	24,88%
Tóxico	31,33%	16,23%	27,92%	24,5%

Tabela 16: Percentagem de menções a entidades dos tipos PESSOA (PER), ORGANIZAÇÃO (ORG), LOCALIZAÇÃO (LOC) e MIS (MISCELÂNEA).

É sabido que as redes sociais online são usadas como um meio de discutir eventos da vida real. Investigamos, também, se nossos dados revelam esse comportamento. Para tanto, mostramos a

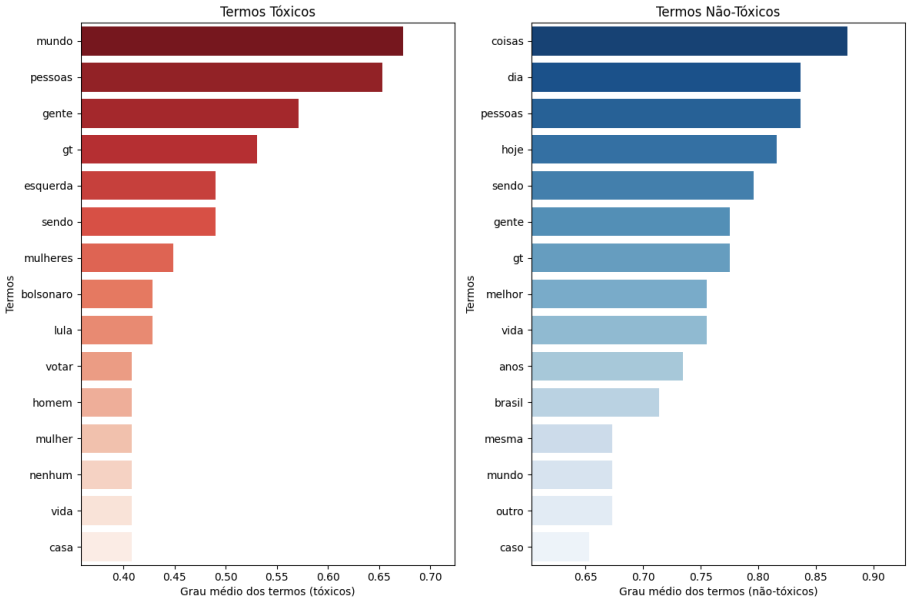


Figura 4: Top-15 termos *tóxicos* e *não-tóxicos* mais influentes extraídos a partir do grafo de coocorrência. No eixo-x, temos o grau médio normalizado dos termos associados com os comentários de cada classe.

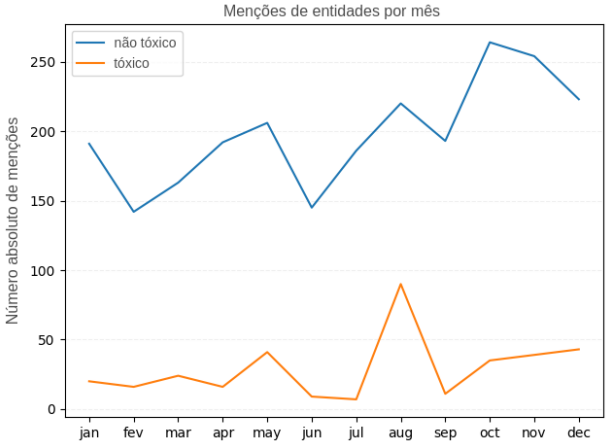


Figura 5: Série temporal mensal de menções de entidades nomeadas.

série temporal mensal dos números de citações de entidades na Figura 5.[6] Há picos significativos no volume de menções em agosto e outubro, que coincidem com o mês de abertura e os dois turnos das eleições brasileiras de 2022. Alguns comentários rotulados como *tóxicos* mencionaram os candidatos presidenciais: *"Vocês são demasiadamente burros! Esse idiota do Bolsonaro pode até "dar um golpe", eu quero ver sustentar esse ato infame, pois, vejamos na década de 60, por exemplo, o Brasil teve essa porcaria de intervenção graças ao apoio do Tio Sam. [..]"*, *"O Lula não vai conseguir ver, pois ele está morto"*.

[6]MISCELÂNEA foi excluída.

4.4. Principais achados

A seguir, resumimos os principais achados do nosso estudo.

4.4.1. Qualidade da anotação

Avaliamos a qualidade do conjunto de dados calculando a concordância entre anotadores, com resultados que corroboram trabalhos semelhantes (Google, 2022b). No entanto, dividimos os anotadores em grupos e nossos resultados mostram que alguns grupos são mais sensíveis a comentários de toxicidade e também apresentam diferentes níveis de qualidade. A forte concordância entre os anotadores do grupo C aponta suas anotações como uma amostra de ouro para avaliar técnicas distintas para o ajuste fino de modelos de aprendizado de máquina de detecção de toxicidade em textos em português brasileiro. Por fim, destacamos a importância da condução de um processo de anotação de dados bem estruturado para a criação de conjuntos de dados de qualidade para o idioma.

4.4.2. Concordância com modelos de aprendizado

Nossa comparação da anotação manual com as classificações da API Perspective mostra que alguns anotadores anotam menor quantidade de comentários tóxicos, enquanto outros são mais sensíveis ao conteúdo *tóxico* gerado. Em geral, a porcentagem média de comentários *tóxicos*

é próxima daquela classificada pela API (entre 10% a 11%). No entanto, a API Perspective é mais sensível a palavrões e não tem o contexto dos tópicos que estão sendo discutidos. Além disso, a API não consegue detectar tipos muito específicos e nuances de ataques direcionados em português (por exemplo, quando grupos específicos são alvo de ofensas na forma de sarcasmo ou ironia). O modelo de linguagem Sabia-2 teve maior correlação com as anotações manuais no geral e também alcançou maior compromisso entre falsos positivos e falsos negativos. O Sabia-2 classificou um percentual de 10,83% de comentários tóxicos —similar ao percentual de comentários tóxicos classificados pelos anotadores—, enquanto o ChatGPT classificou 28,46% dos comentários como *tóxico*. Esse resultado indica um possível ganho de fazer o processo de ajuste fino (*fine-tuning*) em modelos de linguagem usando dados do idioma Português.

4.4.3. Caracterização da linguagem

Os comentários *tóxicos* são, em média, mais longos. Embora tenham uma proporção de etiquetas de POS semelhante à dos *não-tóxicos*, os substantivos e adjetivos mais frequentes apresentam diferenças. Uma clara tendência de aumento nas menções de entidades nomeadas nos subreddits ao longo dos meses, especialmente próximo do período eleitoral brasileiro, mostra o impacto de eventos externos nas interações dos usuários. Isso deve ser considerado ao usar esse conjunto de dados para classificação de textos e criação de modelos, pois o modelo resultante pode ser muito sensível à janela de tempo de dados disponível. Na modelagem de tópicos, identificamos que tópicos em que tivemos discordância total abrangem assuntos relacionados à política, ataques a grupos minorizados e palavrões. Para tópicos em que os comentários foram classificados como *tóxicos* por todos os anotadores, temos a ocorrência de termos relacionados a ideologia, guerra e palavrões. Na análise de bi-gramas, vimos que os termos mais relevantes variam a depender da comunidade (*subreddit*). Entretanto, alguns termos relacionados à política são comuns a todas as comunidades, reforçando a influência dos eventos externos a discussões online. Por fim, a análise de termos influentes usando grafos de coocorrências revelou um padrão similar à análise de bi-gramas, com termos relacionados à política, ideologia e sexo sendo mais influentes no grafo com comentários *tóxicos*, enquanto termos mais genéricos como *Brasil* e *mundo* são mais influentes para comentários classificados como *não-tóxicos*.

Nossos resultados atestam o potencial do nosso conjunto de dados para o ajuste fino de um modelo de aprendizado de máquina em uma tarefa posterior. A alta concordância observada entre os anotadores atesta a consistência dos rótulos. Com os nossos dados, pretendemos fornecer exemplos mais diversificados de textos *tóxicos* de interações de redes sociais online para incentivar o desenvolvimento de modelos de aprendizado de máquina mais robustos, capazes de atenuar comportamentos ofensivos online.

4.4.4. Limitações

Com relação às limitações de nosso estudo, reconhecemos o desafio inerente e a subjetividade da tarefa de rotular conteúdo *tóxico* em um ambiente contextualmente limitado de redes sociais online. Para atenuar esse problema, planejamos repetir o experimento de rotulagem, fornecendo especificamente informações adicionais de contexto para comentários com contexto local ou limitado. Além disso, cabe observar que nosso procedimento de amostragem pode apresentar um viés em relação a tópicos externos específicos que tiveram importância significativa tanto local quanto globalmente durante o período de coleta de dados. Por fim, outro aspecto a se considerar é a diversidade do perfil dos anotadores que participaram do processo de anotação do conjunto de dados. Em experimentos futuros, buscaremos mapear os perfis para garantir maior representatividade dos participantes.

5. Conclusão

Embora os modelos de aprendizado de máquina tenham sido implementados com sucesso como ferramentas de moderação automática para alguns idiomas, ainda não temos suporte para idiomas com poucos recursos, como o português brasileiro. Nosso artigo apresenta um novo conjunto de dados com anotações manuais de comentários tóxicos nas interações de usuários do Reddit dos dez maiores subreddits do Brasil. Nossos resultados indicam uma concordância substancial entre os anotadores e um forte alinhamento com modelos externos pré-treinados para o português, o que apoia a utilização desses dados para tarefas posteriores de aprendizado de máquina.

Em trabalhos futuros, pretendemos integrar esse novo conjunto de dados a modelos de aprendizado de máquina pré-treinados para fornecer ao modelo dados de interações reais em redes sociais. Além disso, pretendemos aproveitar esse conjunto de dados para tarefas mais complexas,

como a detecção de gatilhos de toxicidade em conversas online, a fim de sermos proativos em intervenções de moderação. Por fim, pretendemos realizar uma análise de polaridade para contrastar a caracterização e os resultados obtidos na tarefa de toxicidade.

Agradecimentos

Esta pesquisa foi parcialmente apoiada pelas agências de pesquisa brasileiras CNPq, FAPEMIG e CAPES.

Referências

Almeida, Thales Sales, Hugo Abonizio, Rodrigo Nogueira & Ramon Pires. 2024. Sabiá-2: A new generation of Portuguese large language models. ArXiv:2403.09887 [cs.CL]. 10.48550/arXiv.2403.09887

Almerekhi, Hind, Haewoon Kwak, Bernard J Jansen & Joni Salminen. 2019. Detecting toxicity triggers in online discussions. Em *30th ACM Conference on Hypertext and Social Media*, 291–292. 10.1145/3342220.3344933

Almerekhi, Hind, Haewoon Kwak, Joni Salminen & Bernard J Jansen. 2020. Are these comments triggering? predicting triggers of toxicity in online discussions. Em *Proceedings of The Web Conference (WWW)*, 3033–3040. 10.1145/3366423.3380074

Baumgartner, Jason, Savvas Zannettou, Brian Keegan, Megan Squire & Jeremy Blackburn. 2020. The pushshift Reddit dataset. Em *International AAAI Conference on Web and Social Media*, vol. 14, 830–839. 10.1609/icwsm.v14i1.7347

Benesty, Jacob, Jingdong Chen, Yiteng Huang & Israel Cohen. 2009. Pearson correlation coefficient. Em *Noise reduction in speech processing*, 1–4. Springer Berlin Heidelberg. 10.1007/978-3-642-00296-0_5

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever & Dario Amodei. 2020. Language models are few-shot learners. Em *Neural Information Processing Systems*, 1877–1901.

ElSherief, Mai, Vivek Kulkarni, Dana Nguyen, William Yang Wang & Elizabeth Belding. 2018. Hate lingo: A target-based linguistic analysis of hate speech in social media. Em *International AAAI Conference on Web and Social Media*, vol. 12 1, 10.1609/icwsm.v12i1.15041

Fortuna, Paula, João Rocha da Silva, Leo Wanner & Sérgio Nunes. 2019. A hierarchically-labeled Portuguese hate speech dataset. Em *3rd Workshop on Abusive Language Online*, 94–104. 10.18653/v1/W19-3510

Gillespie, Tarleton. 2020. Content moderation, AI, and the question of scale. *Big Data & Society* 7(2). 10.1177/2053951720943234

Google. 2022a. Perspective API model cards. Acessado em: 10/12/2023.

Google. 2022b. Perspective API training data. Acessado em: 08/04/2023.

Grootendorst, Maarten. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv [cs.CL]. 10.48550/arXiv.2203.05794

Jiang, Jialun Aaron, Morgan Klaus Scheuerman, Casey Fiesler & Jed R Brubaker. 2021. Understanding international perceptions of the severity of harmful content online. *PloS One* 16(8). e0256762. 10.1371/journal.pone.0256762

Kobellarz, Jordan & Thiago Silva. 2022. Should we translate? evaluating toxicity in online comments when translating from Portuguese to English. Em *28th Simpósio Brasileiro de Sistemas Multimídia e Web*, 95–104.

Kukreja, Sanjay, Tarun Kumar, Amit Purohit, Abhijit Dasgupta & Debashis Guha. 2024. A literature survey on open source large language models. Em *7th International Conference on Computers in Management and Business*, 133–143. 10.1145/3647782.3647803

Leite, João Augusto, Diego Silva, Kalina Bontcheva & Carolina Scarton. 2020. Toxic language detection in social media for Brazilian Portuguese: New dataset and multilingual analysis. Em *1st Conference of the Asia-Pacific and the 10th International Joint Conference on Natural Language Processing*, 914–924. 10.18653/v1/2020.aacl-main.91

Nothman, Joel, Nicky Ringland, Will Radford, Tara Murphy & James R Curran. 2013. Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence* 194. 151–175. 10.1016/j.artint.2012.03.006

Oliveira, Amanda S, Thiago C Cecote, Pedro HL Silva, Jadson C Gertrudes, Vander LS Freitas & Eduardo JS Luz. 2023. How good is ChatGPT for detecting hate speech in Portuguese? Em *14th Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, 94–103. 10.5753/stil.2023.233943

de Pelle, Rogers Prates & Viviane P Moreira. 2017. Offensive comments in the Brazilian web: a dataset and baseline results. Em *6th Brazilian Workshop on Social Network Analysis and Mining*, 510–519. 10.5753/brasnam.2017.3260

Petrov, Slav, Dipanjan Das & Ryan McDonald. 2011. A universal part-of-speech tagset. ArXiv [cs.CL]. 10.48550/arXiv.1104.2086

Rademaker, Alexandre, Fabricio Chalub, Livy Real, Cláudia Freitas, Eckhard Bick & Valeria de Paiva. 2017. Universal dependencies for Portuguese. Em *4th International Conference on Dependency Linguistics (DepLing)*, 197–206.

Salehabadi, Nazanin, Anne Groggel, Mohit Singhal, Sayak Saha Roy & Shirin Nilizadeh. 2022. User engagement and the toxicity of Tweets. ArXiv [cs.SI/cs.CY]. 10.48550/arXiv.2211.03856

spaCy. 2022. Portuguese models. Acessado em: 11/04/2023.

Statista. 2022. Number of social media users worldwide from 2017 to 2027. Acessado em: 08/04/2023.

Trajano, Douglas, Rafael Bordini & Renata Vieira. 2023. OLID-BR: offensive language identification dataset for Brazilian Portuguese. *Lang Resources & Evaluation* 58(4). 1263–1289. 10.1007/s10579-023-09657-0

US Congress. 1998. Digital millennium copyright act. *Public Law* 105(304). 112

Vargas, Francielle, Isabelle Carvalho, Fabiana Rodrigues de Góes, Thiago Pardo & Fabrício Benevenuto. 2022. HateBR: A large expert annotated corpus of Brazilian Instagram comments for offensive language and hate speech detection. Em *13th Language Resources and Evaluation Conference (LREC)*, 7174–7183.

Vogels, Emily A. 2021. The state of online harassment. Relatório técnico. Pew Research Center.

Wise, Jason. 2023. Reddit users: how many people use Reddit in 2023? Acessado em: 08/04/2023.

Zannettou, Savvas, Mai ElSherief, Elizabeth Belding, Shirin Nilizadeh & Gianluca Stringhini. 2020. Measuring and characterizing hate speech on news websites. Em *12th ACM Conference on Web Science*, 125–134. 10.1145/3394231.3397902

http://www.linguamatica.com/

linguaMATICA

Artigos de Investigação

Detecção Precoce de Transtornos de Saúde Mental em Português
Bruno Nagamatu & Ivandré Paraboni

Artigos Técnicos

Aperfeiçoando a Hifenização Automática em Português no TEX
Leonardo Araujo & Aline Benevides

PROPOR 2024 | Artigos Convidados

Explorando a Eficácia dos Modelos Generativos em Tarefas de Análise de Sentimentos no Português Brasileiro
Gladson de Araújo, Tiago de Melo & Carlos Figueiredo

Detecção de Posicionamentos a partir de Textos usando Expansão Semiautomática de Córpus
Camila Pereira & Ivandré Paraboni

MultiWOZ-PT: Um Conjunto de Diálogos Orientados a Tarefas em Português
Patrícia Ferreira, Francisco Pais, Catarina Silva, Ana Alves & Hugo G. Oliveira

Explorando Técnicas de Aprendizado em Modelos de Linguagem para Classificação de Discurso de Ódio e Ofensivo em Português
Gabriel Assis et al.

Avaliação Automática do Nível de Complexidade de Textos em Português Europeu
Eugénio Ribeiro, Nuno Mamede & Jorge Baptista

Reconhecimento de Entidades Nomeadas e Vazamento de Dados em Textos Legislativos: Uma Reavaliação da Literatura
Rafael Nunes, André Spritzer, Carla Freitas & Dennis Balreira

Extração de Informação Aberta com LLM para a Língua Portuguesa
Bruno Cabral, Marlo Souza & Daniela Barreiro Claro

RoBERTaLexPT: um modelo RoBERTa jurídico pré-treinado com deduplicação para língua Portuguesa
Eduardo Garcia et al.

Rotulação e Caracterização de Conteúdo Tóxico de Comunidades do Reddit no Brasil
Luiz Quevedo, Ana Pagano, Adriana Pagano & Ana Silva