Proposta recebida em Novembro 2018 e aceite para publicação em Dezembro 2018.

Detecção de Paráfrases na Língua Portuguesa usando Sentence Embeddings

Detecting Paraphrases for Portuguese using Word and Sentence Embeddings

Marlo Souza
Universidade Federal da Bahia
msouza1@ufba.br

Leandro M. P. Sanches
Universidade Federal da Bahia
leandrompsanches@gmail.com

Resumo

A detecção (ou identificação) de paráfrases é a tarefa de determinar se duas ou mais sentenças de comprimento arbitrário possuem o mesmo significado. Os métodos para resolver esta tarefa com potenciais aplicações em sistemas de Processamento de Linguagem Natural. Este trabalho investiga a combinação de diferentes métodos de representação de sentenças em modelos de linguagem por espaços vetoriais e classificadores lineares para o problema de detecção de paráfrases para a língua portuguesa. Os resultados obtidos nesse trabalho estão aquém daqueles obtidos para a tarefa relacionada de detecção de implicação textual na avaliação ASSIN para a língua portuguesa, porém nesse trabalho investigamos a aplicação das representações vetoriais de sentenças para a detecção de paráfrases, outras características usualmente exploradas em sistemas desse tipo podem trivialmente ser incorporadas ao nosso método para melhorar a performance.

Palavras chave

Detecção de Paráfrases, Similaridade Semântica Textual, *Sentence Embeddings*

Abstract

Paraphrase detection/identification is the task of determining whether two or more sentences of arbitrary length possess the same meaning. Methods to solve this task have many potential applications in Natural Language Processing systems. This work investigates the combination of different methods of sentence representation in a vector space model of language and linear classifiers to the problem of paraphrase identification for the Portuguese language. The results obtained in this work are inferior to those obtained for the related task of recognizing textual entailment in the ASSIN evaluation for the Portuguese language, but we point out that in this work we investigate the application of sentence embeddings to the problem of paraphrase detection, as such other features usually explored in systems for this task may be trivially incorporated into our method to improve performance.

Palavras chave

Paraphrase Identification, Semantic Textual Similarity, Sentence Embeddings

1 Introdução

A identificação de paráfrase é a tarefa de determinar se duas ou mais sentenças de comprimento arbitrário possuem o mesmo significado. Para fins desse trabalho, consideraremos uma noção funcional (ou informacional) do que significa duas sentenças terem o mesmo significado. Seguindo as definições de Fonseca et al. (2016) na tarefa ASSIN – Avaliação de Similaridade Semântica e Inferência Textual – ocorrida em 2016, nós consideramos duas sentenças S_1 e S_2 como parafrásticas, se ao ler ambas, uma pessoa conclui que S_1 será verdade se, e somente se, S_2 também o for.

Métodos para detecção de paráfrases possuem aplicações para problemas como Sumarização Automática (Jing & McKeown, 2000), Recuperação de Informação, Sistemas de Resposta a Perguntas (Marsi & Krahmer, 2005), construção automatizada de ontologias (Suresh & Kumar, 2016), entre outros. Não é de se entranhar, portanto, que recentemente muito trabalho tenha sido produzido investigando métodos de identificação de paráfrases e da tarefa relacionada de similaridade semântica textual (*Semantic Textual Similarity* em inglês) (Fonseca et al., 2016; Socher et al., 2011; Yang et al., 2018). Dentre os métodos propostos na literatura, podemos distinguir abordagens baseadas em medir a similaridade lexical, contextual e semântica de sentenças.

Entre os que seguem a última abordagem, recentemente, dada a popularidade da aplicação de representações vetoriais de palavras (*word embeddings*) a várias tarefas de Processamento de Linguagem Natural (PLN), muito trabalho se concentrou no uso de modelos de representação semântica de sentenças através de veto-

DOI: 10.21814/lm.10.2.286

res —comumente chamados de *sentence embeddings*— para detecção de similaridade semântica textual. Uma representação vetorial de sentenças é um modelo de representação que transforma uma sentença de uma determinada linguagem em um vetor em um dado espaço vetorial de alta dimensão. Semelhante a modelos de representações vetoriais de palavras, supõe-se que a geometria do espaço vetorial usado para representar as sentenças codifica aspectos importantes de seu significado.

Representações vetoriais de sentenças foram aplicadas a muitos problemas no Processamento de Linguagem Natural, como Tradução Automática (Bahdanau et al., 2014), Análise de Sentimentos (Kiros et al., 2015), Geração Automática de Diálogos (Yang et al., 2018), etc. Particularmente, para a língua inglesa, os *benchmarks* criados para availar sistemas que medem a similaridade semântica entre sentenças tornaram-se recursos populares para avaliar a qualidade de modelos de representação vetorial de palavras (*word embeddings*) e de sentenças (*sentence embeddings*). Um exemplo de tais *benchmarks* é o conjunto SICK (Marelli et al., 2014) para similaridade semântica entre textos.

Este trabalho consiste de uma versão estendida do trabalho "Detecting Paraphrases for Portuguese using Word and Sentence Embeddings" apresentado no POP 2018 —*1st Workshop on Linguistic Tools and Resources for Paraphrasing in Portuguese* ocorrido conjuntamente com o PROPOR 2018 na cidade de Canela no Brasil. Nele, apresentamos investigações iniciais da aplicação de métodos de representações vetoriais de sentenças para identificação de paráfrase para a língua portuguesa. Em relação à publicação original, nós apresentamos aqui algumas avaliações posteriores realizadas para responder a questionamentos surgidos nas discussões do evento, como testar o efeito da calibração de parâmetros experimentais assim como do uso de diferentes modelos de representações vetoriais de sentenças nos resultados obtidos.

O presente trabalho está organizado da seguinte forma. Na Secção 2, discutimos alguns dos métodos de representação vetorial de sentenças utilizados nesse trabalho; na Secção 3, apresentamos os trabalhos relacionados ao nosso, i.e. trabalhos que tratam sobre deteção automática de paráfrases, com um foco na apresentação daqueles que utilizam o mesmo corpus que o nosso em seus experimentos; na Secção 4, descrevemos nosso trabalho experimental e discutimos os resultados obtidos; finalmente, na Secção 5, apresentamos algumas considerações finais.

2 Representações Geométricas de Palavras e Sentenças

Modelos de representação vetorial de palavras são modelos de linguagem que exploram a similaridade distribucional entre palavras em um grande corpus para aprender representações de palavras de uma linguagem como vetores em um dado espaço vetorial de alta dimensão. Da mesma forma, modelos de representação vetorial de sentenças visam codificar sentenças como vetores em um determinado espaço vetorial de forma a representar, na geometria do espaço vetorial, o significado original da sentença.

Dado um modelo de representação vetorial de palavras, podemos construir modelos simples para representação de uma sentença através da *agregação* das representações das palavras que a compõem. Tal método, apesar de simplório em primeira análise, pode ser justificado através do princípio da composicionalidade de significados, que afirma que o significado de uma sentença é obtido por alguma transformação no significado de seus constituintes. Assim, métodos seguindo essa abordagem (Mihalcea et al., 2006; Conneau et al., 2018) visam estabelecer alguma transformação que realiza tal *agregação*, i.e. de forma a codificar o significado de uma sentença a partir do significado das palavras que a constituem. De uma forma geral, métodos baseados em agregação buscam representar como o significado de palavras individuais contribuem para o significado da sentença.

Uma maneira trivial de fazê-lo é tomar o centroide da representação vetorial de todas as palavras (ou pelo menos das palavras lexicais) que constituem uma sentença como sua representação. Isso corresponde à ideia de que cada palavra contribui igualmente para determinar o significado da sentença. Essa representação vetorial pode ser obtida tomando a média dos vetores representando todas as palavras que compõem a sentença.

Não está claro, entretanto, que cada palavra contribui igualmente para o significado da sentença. De fato, algumas palavras podem atuar como marcadores gramaticais na sentença e seu significado individual pode não contribuir para o significado da frase, e.g. o caso da palavra *pas* que foi gramaticalizada na negação verbal "*ne ... pas*" ("não") em francês. Para explicar a diferença na importância de cada palavra para o significado da sentença, a representação da frase pode ser tomada como a *agregação ponderada* do vetor de cada palavra.

Muitas estratégias diferentes de ponderação podem ser estabelecidas de forma a levar em consideração a estrutura da frase ou as propriedades distributivas das palavras. Uma abordagem comum, semelhante à abordagem de Mihalcea et al. (2006) para calcular a similaridade de sentenças, é tomar o Inverso da Frequência nos Documentos (IDF, do inglês *Inverso Document Frequency*) de cada palavra em um dado corpus representativo como uma medida de importância para a palavra. A ideia fundamental de tal abordagem é que as palavras menos comuns da língua contribuam mais —ou tenham alguma *saliência*— no significado da sentença.

Observe que a maioria dos modelos de representação vetorial de palavras visa capturar padrões de co-ocorrência de palavras presentes no corpus de treinamento. No entanto, a presença de palavras fora de contexto pode causar ruído no modelo treinado (Arora et al., 2017). Assim, o método de agregação de representações de palavras para calcular a representação da sentença pode resultar num acúmulo de ruído e, portanto, degradar o significado da sentença em sua representação vetorial. Para superar esse problema, Arora e colegas propõem o uso de métodos de fatoração de matriz para identificar o componente principal dos vetores de palavra, que é interpretado como o *ruído acumulado* na representação da sentença por agregação. Tal ruído é, então, eliminado da representação da sentença. Essa técnica é conhecida como *Smooth Inverse Frequency* (SIF, ou Frequência Inversa Suave).

Trabalhos como o de Kiros et al. (2015), por outro lado, visam aplicar métodos de aprendizagem de máquina para aprender a representação de uma sentença a partir de seus padrões de distribuição em um grande corpus, similar aos métodos para aprendizagem de representações vetoriais de palavras. Esses métodos geralmente se baseiam nas representações das palavras que constituem uma sentença e tentam aprender com o corpus a melhor maneira de agregar tais representações para calcular a representação das sentenças. Recentemente, muitos métodos diferentes para aprender representações vetoriais de sentenças foram propostos na literatura, geralmente empregando redes neurais profundas e recorrentes para aprender tais representações (Conneau et al., 2017; Kiros et al., 2015; Le & Mikolov, 2014; Patro et al., 2018; Socher et al., 2011). Esses métodos foram aplicados com sucesso em diversas tarefas de Processamento de Linguagens Naturais (Cer et al., 2018; Howard & Ruder, 2018; Logeswaran & Lee, 2018).

Particularmente interessante para nós, é o método de Skip-Thought proposto por Kiros et al. (2015). Skip-Thought é um método não supervisionado de aprendizagem de representações vetoriais de sentenças usando uma arquitetura codificador-decodificador de redes neurais para prever a vizinhança de uma certa sentença, similar ao método *Skip-Gram* para aprender representações vetoriais de palavras. O impacto de tal trabalho reside no fato do mesmo descrever um método não-supervisionado para aprendizagem de tal representação e, portanto, não requerem dados anotados para seu treinamento. Métodos supervisionados para representações de sentenças, como InferSent (Conneau et al., 2017), por outro lado, provaram ser bem-sucedidos para aplicações específicas, mas vêm com o preço de depender de dados anotados – que podem não estar disponíveis para todos os idiomas.

Recentemente, Cer et al. (2018) propuseram o codificador universal de sentenças (*Universal Sentence Encoder*, em inglês), um método para aprender representações vetoriais de sentenças baseado em redes neurais que, de acordo com os autores, gera representações de uso geral para serem aplicadas em diversas tarefas de PLN. Os autores aplicam seus codificadores universais para cinco tarefas distintas: análise de sentimentos, detecção de subjetividade, classificação de questões, similaridade semântica textual e testes de associação implícita. Os autores apresentam resultados positivos para o uso de representações vetoriais de sentenças em tarefas de PLN, especialmente para os casos de baixa disponibilidade de dados.

Neste trabalho, focaremos na aplicação de diferentes modelos de representação de sentenças, e nas medidas de similaridade semântica relacionadas a esses modelos, ao problema de identificação de paráfrase para a língua português. Em certo sentido, nosso trabalho é semelhante ao de Feitosa & Pinheiro (2017), ou de Fialho et al. (2016), que avaliam o uso de algumas medidas de similaridade para o problema da similaridade semântica textual.

3 Identificação Automática de Paráfrases

Trabalhos sobre identificação de paráfrase podem ser divididos em três grandes categorias. Primeiro, há os trabalhos baseados em heurísticas, como medidas de semelhança semântica e tesauros ricamente anotados, como os trabalhos de Cordeiro et al. (2007) e Fernando & Stevenson (2008). Outros trabalhos, como o de Shinyama

et al. (2002), computam semelhanças contextuais entre palavras, como co-ocorrência em uma frase ou sintagmas, e exploram tais semelhanças para detectar semelhanças de significado entre duas sentenças – geralmente aplicando algoritmos de aprendizado de máquina para identificar as paráfrases. Finalmente, a terceira categoria de métodos se baseia em princípios de semântica distribucional, como a hipótese distribucional[1]

O trabalho de Cordeiro et al. (2007) propõe uma métrica para calcular a semelhança de significados entre duas sentenças, baseada na sobreposição de unidades lexicais. Observe que trabalhos que usam medidas de variação lexical/estrutural para identificar paráfrases, como o de Dolan et al. (2004) que usa a distância de Levenshtein entre duas sentenças, são capazes de identificar apenas aqueles exemplos em que as sentenças têm estrutura quase idêntica. Enquanto o trabalho de Cordeiro et al. (2007) evita muitas dessas armadilhas, uma vez que não se baseia na estrutura da sentença, como a sobreposição lexical é uma condição bastante restritiva para identificar paráfrases, sua abordagem é limitada no sentido de que não pode detectar paráfrases nas quais há variação significativa nas descrições de entidades e ações nas sentenças, como o uso de nomes diferentes e descrições definidas[2]. Considere, por exemplo, as sentenças "*Lula foi libertado nesta madrugada*" e "*O presidente Luís Inácio da Silva foi solto no início desta manhã.*" As frases claramente possuem significado semelhante, porém apresentam baixa sobreposição lexical.

Trabalhos como o de Mihalcea et al. (2006) e de Fernando & Stevenson (2008), por outro lado, propõem a exploração de medidas de similaridade entre sentenças para identificar paráfrases na língua inglesa, baseadas não somente na sobreposição lexical e semelhança estrutural, mas também em similaridades semânticas, contextuais e distributivas. Esses trabalhos exploram informações ricas com base em dicionários de sinônimos anotados, como a WordNet (Miller, 1995), e grandes corpora, como explorados por Turney & Littman (2002). Eles são flexíveis no sentido de que podem ser empregados usando diferentes medidas de similaridade, que exploram recursos linguísticos ricamente anotados ou grandes corpora não-anotados disponíveis para uma determinada linguagem.

Socher et al. (2011) empregam autocodificadores recursivos (RAE, *Recursive AutoEncoders*), um tipo de rede neural profunda não-supervisionada seguindo o modelo codificador-decodificador, para codificar a estrutura sintática das sentenças. Essas representações são, então, aplicadas para medir a semelhança de duas sentenças a nível de palavras e de seus constituintes sintáticos (sintagmas), que são então usadas para treinar um classificador de paráfrase.

Da mesma forma, Yin & Schütze (2015) propõem o uso de redes neurais convolucionais profundas para resolver o problema da detecção de paráfrase. Eles propõem uma nova arquitetura de rede neural que, segundo eles, permite codificar múltiplos níveis de granularidade do significado das sentenças. Essas representações são então usadas para treinar um classificador logístico para identificar paráfrases.

Esses trabalhos mais recentes são semelhantes ao de Mihalcea et al. (2006) e de Fernando & Stevenson (2008) por também explorarem a similaridade da distribuição de palavras e sintagmas em grandes corpora não anotados para calcular similaridade semântica entre sentenças. A diferença entre esses trabalho reside no fato das abordagens mais recentes considerarem a estrutura sintática da sentença para calcular a semelhança semântica entre elas, enquanto aqueles anteriores não levam essa informação em consideração.

O trabalho de Kiros et al. (2015) descreve um modelo de aprendizado não supervisionado de um codificador de sentença genérico, que pode ser aplicado a diferentes tarefas subsequentes de PLN. Semelhante ao que é feito para modelos de representação de palavras, os autores treinam uma arquitetura codificador-decodificador que tenta reconstruir as sentenças circundantes de uma passagem codificada. Os autores avaliam os modelos gerados pelo seu método em 8 tarefas: semelhança semântica, detecção de paráfrase, ranking de sentenças e imagens, classificação de tipo de pergunta e análise de sentimento e subjetividade.

Nosso trabalho decorre dos mais recentes que aplicam redes neurais e modelos de espaço vetorial para representar a informação semântica expressa em uma sentença. Nosso objetivo é avaliar sua utilidade para o problema de detecção de paráfrase para a língua portuguesa.

[1] A hipótese distribucional afirma que itens linguísticos com distribuições estatísticas semelhantes em grandes corpora têm significados semelhantes (Sahlgren, 2008).

[2] Note que essa crítica se justifica para a noção de paráfrase adotada nesse trabalho. Devemos salientar, entretanto, que noutra perspectiva de paráfrase, como a adotada pelo trabalho de Baptista nesse volume, tal abordagem está bem justificada, uma vez que duas sentenças são ditas parafrásticas quando "há uma equivalência *transformacional* entres sentenças que requer que o mesmo material lexical significativo esteja envolvido" (Baptista, 2018, tradução nossa).

Outros trabalhos sobre a detecção de paráfrase para o português foram realizados, especialmente no contexto da tarefa de avaliação conjunta ASSIN – Avaliação de Similaridade Semântica e Inferência Textual (Fonseca et al., 2016). Embora alguns desses trabalhos empreguem características obtidas com modelos de representação vetorial de palavras, mais notavelmente o trabalho de Hartmann (2016) para o problema de similaridade semântica, até onde sabemos, nenhum deles avaliou o uso diferentes métodos de representação de sentenças para esse problema.

O trabalho de Fialho et al. (2016) apresenta o método INESC-ID que utiliza informações diversas métricas de similaridade e sobreposição textual típicas da área de tradução automática para o problema de Inferência em Linguagem Natural ou Reconhecimento de Implicação Textual (RTE do inglês*Recognizing Textual Entailment*). Essa é uma tarefa mais geral que a detecção de paráfrase e, de fato, a engloba. Os autores relatam uma performance de 0.64 e 0.66 de medida F1 na tarefa RTE sobre o corpus ASSIN para as variantes europeia e brasileira da língua, respectivamente. Note entretanto que, como esses autores não avaliaram separadamente a performance do seu sistema na detecção de paráfrases, não podemos comparar nossos resultados com o deles.

Barbosa et al. (2016) apresentam o sistema Blue Man Group para o problema de RTE utilizando classificadores treinados sobre vetores de características obtidos usando a cosntrução de redes semânticas entre as palavras das duas sentenças e atributos de nível textual, baseados no trabalho de Kenter & De Rijke (2015). Os autores relatam uma medida F1 de 0.58 para o problema de RTE, mas não avaliam a performance de seu sistema sobre detecção de paráfrases separadamente.

O trabalho de Feitosa & Pinheiro (2017) avalia a aplicação de diversas medidas de similaridade textual - baseadas em aspectos sintáticos e semânticos do texto - à tarefa de RTE utilizando o corpus do ASSIN. Os autores relatam uma medida F1 de 0.71 em seus experimentos para a tarefa RTE, porém não avaliam seus resultados no reconhecimento de paráfrases somente e, portanto, seus resultados não podem ser comparados aos nossos.

O trabalho de Rocha & Lopes Cardoso (2018) utilizam classificadores multi-classe considerando características lexicais, sintáticas e semânticas de textos para a tarefa de RTE sobre o corpus ASSIN. Esses autores apresentam um resultado de 0.71 de (Macro) F1 para a tarefa de RTE. Enquanto os autores não apresentam os resultados de seu método para detecção de paráfrase sobre o conjunto de dados de teste, eles apresentam uma avaliação sobre o conjunto de treino usando validação cruzado, para o qual obtém uma medida F1 de 0.6 para a variante europeia do corpus e 0.52 para o corpus de treino com ambas as variantes combinadas.

O trabalho de Fonseca & Aluísio (2018) explora o uso de diferentes informações sintáticas para a tarefa de RTE também utilizando o corpus do ASSIN e alcança resultados 0.72 de medida F1. Esses autores também não apresentam seus resultados discriminando a performance no seu método para detecção de paráfrases, de modo que não podemos comparar nossos resultados com os deles.

Note que alguns trabalhos recentes tratando do problema de determinação de similaridade semântica entre textos na língua portuguesa também utilizam o corpus do ASSIN. É importante pontuar, entretanto, que enquanto os problemas de similaridade semântica e detecção de paráfrases são certamente relacionados, não é claro que um possa ser reduzido ao outro. Alguns desses trabalhos recentes, por exemplo (Silva et al., 2017; Pinheiro et al., 2017; Gonçalo Oliveira et al., 2017; de Barcelos Silva & Rigo, 2018; Alves et al., 2018), fornecem evidências de quais tipos de características linguísticas de um texto, como informação sintática, sobreposição lexical, etc. podem ser utilizadas também por sistemas de detecção de paráfrase.

Nesse trabalho, entretanto, nos concentraremos no uso das representações vetoriais de sentença – e as informações que podemos derivar com as mesmas como medidas de similaridade entre essas representações – para a detecção de paráfrases, sem considerar características de outras naturezas. A razão para tal escolha se recai no fato que estamos interessados em investigar quanta informação sobre o conteúdo da sentença pode ser codificado em sua representação vetorial.

4 Usando Representações de Sentenças para Detecção de Paráfrases

Nesta secção, descrevemos a implementação de classificadores de paráfrase que recebem duas sentenças e decidem se estas são parafrásticas. Investigamos diferentes classificadores lineares treinados em dados de representação das sentenças e similaridades obtidos com o uso de quatro diferentes formas de representação de sen-

tença. Abaixo, descrevemos os dados que usamos em nossos experimentos, bem como os resultados obtidos em nossa investigação.

4.1 Dados

Neste trabalho, usamos três fontes de dados principais: um modelo de representação vetorial de palavras para o português, um corpus não anotado de textos em português para treinar o modelo Skip-Thought e o corpus ASSIN (Fonseca et al., 2016) para treinar e avaliar nossa classificadores.

Para o modelo de representação de palavras usado em nossos experimentos, optamos por usar o modelo FastText (Bojanowski et al., 2016) pré-treinado para a língua portuguesa da Facebook Research[3], que foi treinado no corpus de artigos da Wikipédia escritos em português. Estamos cientes da existência de outros modelos de *word embeddings* para a língua portuguesa que estão disponíveis para uso, particularmente aqueles no Repositório de *Word Embeddings* do NILC[4] analisados no trabalho de Hartmann et al. (2017). Escolhemos o modelo FastText da Facebook Research, entretando, por dois motivos simples: primeiramente, o FastText se tornou um dos modelos de melhor desempenho de representação de palavras na literatura, veja por exemplo os experimentos de Hartmann et al. (2017); segundo, os tamanhos dos modelos NILC de maior dimensionalidade são simplesmente muito grandes para os recursos computacionais disponíveis para nós, enquanto o modelo da Facebook tem uma dimensionalidade competitiva, embora ainda tenha um tamanho gerenciável que nos permite realizar nossos experimentos. De qualquer forma, na Subsecção 4.4, nós testamos o impacto do modelo utilizado nesses experimentos.

O corpus utilizado para treinar o método Skip-Thought é composto por 10.354.228 sentenças e 308.261.905 *tokens*. O corpus foi compilado tomando os artigos escritos em português da Wikipédia, um extrato de cerca de 1000 documentos do corpus de textos jornalísticos PLN-BR Full (Bruckschen et al., 2008) e cerca de 700 resenhas de filmes dos sites *CinePlayers* [5] e *Cinema com Rapadura*[6]. Escolhemos complementar o corpus da Wikipédia com novos documentos com a principal finalidade de aumentar a robustez do modelo treinado, dado o treinamento de um modelo neural como o Skip-Thought requer um grande quantidade de dados de treino. A escolha dos textos utilizados para compor esse corpus se deu pela imediata disponibilidade dos mesmos para que pudéssemos utilizá-los, assim como para garantir uma diversidade de estilos representados no corpus – textos enciclopédicos, jornalísticos e opinativos.

Para computar as representações por agregação ponderada, bem como a representação SIF, também foi utilizado um dicionário de valores IDF para palavras na língua portuguesa – tanto as variantes do português brasileiro quanto do português europeu – composto por 873.329 unidades lexicais. Este dicionário foi obtido processando uma fração do corpus usado para treinar o modelo Skip-Thought aleatoriamente selecionada. Por limitação de tempo e recursos computacionais, não pudemos realizar o cálculo de IDF para todas as palavras do corpus. Assim, preferimos selecionar um extrato do corpus e calcular esses valores.

Para treinar os classificadores, utilizou-se o fragmento de treino do corpus ASSIN (Fonseca et al., 2016) de semelhança textual e paráfrases. Tal corpus é composto por 5000 pares de frases anotadas com similaridade entre sentenças e relações de inferência textual, dentre os quais 295 são exemplos anotados de pares de sentenças parafrásticas. Os classificadores foram avaliados no fragmento de teste do mesmo corpus, contendo 4000 pares de sentenças, dos quais 239 são exemplos positivos de paráfrases. Ficou reservado o fragmento de desenvolvimento, composto por 1000 pares de sentenças, das quais 70 são exemplos positivos de paráfrase, para avaliação de parâmetros experimentais, que são apresentados na Subsecção 4.4.

4.2 Projeto Experimental

Para avaliar a aplicação de modelos de representação vetorial de sentenças ao problema da detecção de paráfrase em português, treinamos um modelo Skip-Thoughts para a língua portuguesa e aplicamos esse modelo, juntamente com o modelo FastText do Facebook. Neste experimento, utilizamos a implementação do método FastText da biblioteca Gensim 3.5[7] e empregamos a representação centroide (média dos vetores de palavras), a agregação ponderada baseada na medida IDF, a representação SIF e a representação Skip-Thought de sentenças.

Note que, na literatura relacionada, existem duas formas principais de representar vetores de pares de sentenças para determinar se são pa-

[3] https://research.fb.com/fasttext/
[4] http://nilc.icmc.usp.br/embeddings
[5] http://www.cineplayers.com
[6] http://cinemacomrapadura.com.br
[7] https://radimrehurek.com/gensim/

rafrásticas ou não. Trabalhos como o de Socher et al. (2011) e Yin & Schütze (2015) utilizam diretamente as representações vetoriais $\overrightarrow{u}$ e $\overrightarrow{v}$ para as sentenças s_1 e s_2 como entrada para os classificadores, enquanto trabalhos como o de Kiros et al. (2015) usam combinações desses vetores pelo produto componente-a-componente $\overrightarrow{u} \cdot \overrightarrow{v}$ e a diferença entre os vetores $\overrightarrow{u} - \overrightarrow{v}$ indicando a semelhança e diferença semântica entre ambos, respectivamente. Nesse trabalho exploraremos ambas as formas de representar o conteúdo da sentença.

Dessa forma, processamos os dados e obtivemos um conjunto de dados diferente para cada método de representação de sentença contendo as seguintes características (*features*):

1. a representação vetorial $\overrightarrow{u}$ da primeira sentença do par;
2. a representação vetorial $\overrightarrow{v}$ da segunda sentença do par;
3. o produto componente a componente entre os vetores $\overrightarrow{u}$ and $\overrightarrow{v}$, i.e. o vetor $\overrightarrow{u} \cdot \overrightarrow{v}$;
4. a norma do vetor $\overrightarrow{u} \cdot \overrightarrow{v}$;
5. a diferença vetorial entre os vetores $\overrightarrow{u}$ e $\overrightarrow{v}$, i.e. $\overrightarrow{u} - \overrightarrow{v}$;
6. a norma do vetor $\overrightarrow{u} - \overrightarrow{v}$;
7. o cosseno entre as representações vetoriais das duas sentenças;

Note que, normalmente, considera-se que o cosseno entre dois vetores que representam sentenças codifica alguma forma de similaridade semântica entre elas. Assim, criamos também um conjunto de dados diferente que contendo somente os valores de similaridade para cada par de sentenças no corpus usando todos os diferentes métodos de representação de sentença investigados neste trabalho. Queremos com tal conjunto de dados avaliar se a similaridade entre as sentenças pode ser usada como um indicador de paráfrase. Também agregamos todas as informações em um único conjunto de dados, no qual cada ponto é composto de todas as informações obtidas para cada método de representação. Queremos avaliar com este conjunto de dados se diferentes representações podem codificar diferentes aspectos do significado das sentenças e se esses diferentes aspectos podem ser compostos para identificar paráfrases.

Avaliamos os classificadores obtidos usando as métricas bem estabelecidas de: Precisão (Prec), computada como a porcentagem de exemplos corretos de paráfrases dentro daqueles que foram identificados pelo sistema como exemplos parafrásticos; Cobertura (Rec, do inglês *Recall*), computada como a porcentagem dos exemplos corretamente identificados como parafráses pelo sistema dentro de todos os exemplos parafrásticos no corpus de treino; e F1, média harmônica entre a Precisão e a Cobertura (Alpaydin, 2009).

4.3 Resultados

Treinamos diferentes classificadores usando dados obtidos com cada representação de sentença. Na Tabela 1, apresentamos os resultados obtidos para cada classificador explorado neste trabalho, ou seja, Máquinas de Vetor de Suporte (SVM, do inglês *Support Vector Machines*), Naïve Bayes (NB), e Árvores de Decisão usando o algoritmo J48 (AD). Tais classificadores foram treinados em dados obtidos por cada método de representação de sentenças, ou seja, o centroide dos vetores das palavras, i.e. sua média (Avg, do inglês *Average*), a agregação ponderada de vetores de palavras (Agg), a representação SIF (SIF) e a representação Skip-Thought (ST). Treinamos ainda os classificadores em um conjunto de dados contendo apenas os valores de similaridade obtidos (Sim) e um outro contendo todas as informações combinadas (Total).

Nestes experimentos, utilizamos a biblioteca SciKit-Learn [8] para linguagem Python para a implementação dos classificadores utilizados nesse trabalho e das técnicas de balanceamento de dados discutidas na Subsecção 4.4, assim para o cálculo das métricas de Precisão, Cobertura e F1.

Como os dados são severamente desbalanceados entre as classes, nós também avaliamos o impacto do balanceamento dos dados no desempenho dos classificadores. Para balanceamento dos dados, nós utilizamos a técnica de *oversampling* por amostragem aleatória nos dados. Os resultados dos classificadores treinados sobre esse conjunto balanceado de dados é exibido na Tabela 2.

Nos dados não balanceados, o classificador Naïve Bayes parece ter um comportamento marginalmente melhor (e mais estável) que os outros, entretanto não é possível afirmar que existem diferenças significativas na performance. Os métodos de representação com melhor desempenho foram os método de centroide (Avg), de similaridades (Sim) e o método com informações combinadas (Total). Para os dados balanceados, o classificador baseado em Máquinas de Vetor de Suporte (SVM) tem uma performance ligeiramente superior, porém ainda similar aos outros. Sobre esses dados, novamente as representações por centroide (Avg), similaridade (Sim) e in-

[8] https://scikit-learn.org/

formações combinadas (Total) alcançaram os melhores resultados.

Método	Classificador	Métricas		
		Prec	Rec	F1
Avg	SVM	0.38	0.19	0.25
	NB	0.20	0.72	**0.31**
	AD	0.21	0.24	0.22
Agg	SVM	0.13	0.04	0.05
	NB	0.10	0.63	0.17
	AD	0.11	0.14	0.12
SIF	SVM	0	0	0
	NB	0.09	0.71	0.15
	AD	0.10	0.10	0.10
Skip	SVM	0.20	0.06	0.09
	NB	0.06	0.94	0.12
	AD	0.11	0.12	0.11
Sim	SVM	0.50	0.08	0.13
	NB	0.13	0.90	0.22
	AD	0.25	0.26	0.25
Total	SVM	0.29	0.31	0.30
	NB	0.09	0.81	0.15
	AD	0.29	0.32	0.30

Tabela 1: Resultados da avaliação dos classificadores treinados para identificação de paráfrases

Método	Classificador	Métricas		
		Prec	Rec	F1
Avg	SVM	0.21	0.44	0.29
	NB	0.19	0.72	0.30
	AD	0.24	0.23	0.23
Agg	SVM	0.10	0.31	0.15
	NB	0.09	0.66	0.17
	AD	0.09	0.10	0.09
SIF	SVM	0.09	0.43	0.16
	NB	0.08	0.71	0.15
	AD	0.10	0.13	0.11
Skip	SVM	0.09	0.30	0.14
	NB	0.06	0.94	0.12
	AD	0.13	0.14	0.13
Sim	SVM	0.21	0.77	**0.33**
	NB	0.09	0.94	0.17
	AD	0.28	0.28	0.28
Total	SVM	0.27	0.33	0.30
	NB	0.09	0.81	0.16
	AD	0.30	0.31	0.31

Tabela 2: Resultados da avaliação de classificadores treinados sobre o conjunto de dados balanceados

Note que, em comparação com os resultados originais (Souza & Sanches, 2018), percebemos claramente uma melhora na performance dos classificadores SVM e AD, assim como da representação por informações globais. Atribuímos esse fato ao aumento na quantidade de dados de treino ao utilizar o corpus ASSIN completo, não somente a variante do Português Brasileiro como naquele trabalho[9]. Isso indica que, pelo fato da representação com informações combinadas gerar um espaço de representação com grande dimensionalidade, os resultados originais para tal representação podem ter sofrido por esparsidade de dados.

É importante salientar que a melhoria dos resultados ao utilizar ambas as variantes é relativamente surpreendente pois, enquanto ambas as variantes se comportam de forma similar para a tarefa de similaridade semântica, os participantes da ASSIN verificaram sistematicamente diferenças entre o comportamento dos sistemas nas duas variantes para a tarefa de RTE. Particularmente, Rocha & Lopes Cardoso (2018) argumenta que as características de implicação e paráfrase parecem ser diferentes em ambos conjuntos de dados.

4.4 Avaliação de Parâmetros Experimentais

É importante notar que o desempenho das técnicas investigadas neste trabalho está claramente abaixo do desempenho relatado para o idioma inglês (c.f. (Kiros et al., 2015), por exemplo) ou para a inferência textual relatada pelos concorrentes no desafio ASSIN (c.f. (Barbosa et al., 2016) ou (Fialho et al., 2016)). As razões para esse baixo desempenho podem surgir de inúmeros parâmetros experimentais utilizados, como o modelo de *word embeddings* adotado ou a técnica de balanceamento de dados utilizada nos experimentos. Para avaliar o efeito desses parâmetros experimentais, realizamos novos experimentos variando-os e comparando a performance do modelo. É importante salientar que nos experimentos discutidos nessa subsessão, usamos como corpus de treino o fragmento de treino do corpus ASSIN (Fonseca et al., 2016), como nos experimentos anteriores, e para teste, utilizamos o fragmento de desenvolvimento (*dev*) do mesmo corpus.

[9]Nesse ponto, agradecemos ao revisor por sua contribuição ao pontuar que a utilização dos dados do ASSIN para a variante europeia da língua poderia melhorar os resultados, como de fato foi observado.

Note que nos experimentos discutidos anteriormente foi utilizada uma técnica ingênua de balanceamento de dados por amostragem aleatória. Apesar do balanceamento dos dados ter apresentado um efeito positivo na performance de alguns classificadores (compare as Tabelas 1 e 2), o uso de tal técnica pode ter resultado num sobreajuste (*overfitting*) dos classificadores nos dados de treino, o que explicaria os baixos valores de Precisão obtidos. Existem na literatura, entretanto, técnicas mais avançadas de balanceamento de dados por sintetização de exemplos, como o SMOTE (Chawla et al., 2002) e o ADASYN (He et al., 2008). Nós decidimos, então, avaliar se o uso de tais técnicas pode melhorar a performance dos classificadores. Para avaliar tal o impacto, usamos o classificado Naïve Bayes e a representação de centroide (Avg), que obtiveram os melhores resultados nos experimentos apresentados anteriormente. Os resultados são apresentados na Tabela 3.

Técnica	Métricas		
	Prec	Rec	F1
Amostragem	0.23	0.79	0.36
SMOTE	0.37	0.54	**0.44**
ADASYN	0.34	0.54	0.42

Tabela 3: Resultados da avaliação do impacto de uso de técnicas de balanceamento de dados

Enquanto em valores absolutos, os resultados apresentados na Tabela 3 indicam que os métodos mais sofisticados de *oversampling* ocasionaram em classificadores com maior Precisão que a simples amostragem aleatória, a diferença entre a performance dos mesmos não parece ser estatisticamente significante. O que é possível observar, entretanto, é que os métodos SMOTE e ADASYM obtém maior precisão, provavelmente devido a uma melhor generalização sobre os dados de treino.

Um importante ponto a se considerar nesse resultado, entretanto, é o fato que a diferença de performance entre os métodos é mais pronunciada se considerarmos somente os dados da variante do Protuguês Brasileiro do corpus ASSIN (F1 de 0.24 para Amostragem Aletória, contra F1 de 0.37 para SMOTE), provavelmente pela menor quantidade e variedade de dados. Isso indica que o uso da técnica de *oversampling* por amostragem aleatória nos experimentos originais, publicados em (Souza & Sanches, 2018), pode ter um importante impacto nos resultados obtidos - devido a potencial sobreajuste dos classificadores.

Outra possível razão para o baixo desempenho dos classificadores testados pode ser a falta de robustez do modelo de *word embeddings* adotado. Tal modelo foi treinado no corpus de artigos da Wikipedia – um corpus pequeno para aprendizado não supervisionado deste tipo de modelos. Para avaliar o impacto do modelo de *word embeddings* usado, realizamos novos experimentos com os modelos treinados por Hartmann et al. (2017) usando os métodos FastText (Bojanowski et al., 2016), Word2Vec (Mikolov et al., 2013) e GloVe (Pennington et al., 2014) com 300 dimensões. Apesar da dimensionalidade dos modelos testados ser igual a do modelo usado originalmente, os modelos de Hartmann et al. (2017) foram treinados sobre um conjunto de dados muito maior que aquele da Facebook Research (FB Fasttext). Os resultados são apresentados na Tabela 4, que descreve as métricas obtivas pelo classificador Naïve Bayes utilizando o método de representação pelo centroide (Avg) com balanceamento usando a técnica SMOTE. Assim como nos experimentos anteriores, utilizamos as implementações da biblioteca Gensim 3.5[10] para os métodos Word2Vec, FastText e GloVe.

Modelo	Métricas		
	Prec	Rec	F1
FB FastText	0.37	0.54	0.44
FastText	0.37	0.57	**0.45**
Word2Vec	0.29	0.60	0.39
GloVe	0.30	0.41	0.35

Tabela 4: Resultados da avaliação do uso de diferentes modelos de *word embeddings*

Podemos perceber que, apesar dos modelos de Hartmann et al. (2017) serem treinados sobre um conjunto de dados maior que o do modelo da Facebook Research usados em nosso experimentos, os resultados apresentados na Tabela 4 não fornecem evidências de que o uso de modelos diferentes impactem significativamente a performance do classificador.

Por fim, perceba na Tabela 2 os valores de similaridade entre as sentenças demonstraram-se como um poderoso indicador de paráfrase.

É importante perceber, entretanto, que enquanto os classificadores treinados sobre os dados de similaridade apresentaram um valor de Cobertura (*Recall*) bastante elevado, os valores de Precisão obtidos foram bastante baixos. Uma possível explicação para tal fenômeno é que no corpus ASSIN, algumas sentenças possuem alto valor de similaridade entre si, porém não constituem exemplo de paráfrase. Tais sentenças podem se constituir em ruído para os classificado-

[10] https://radimrehurek.com/gensim/

res e impactar a performance. Para avaliar a influência de tais sentenças ruidosas, nós avaliamos o impacto de descartar do conjunto de treino todos os exemplos de sentenças que possuem um valor de similaridade acima de um determinado limiar e que não sejam para parafrásticas. Para identificar o impacto desse valor de limiar nos resultados, testamos os limiares no intervalo entre 1.5 e 5.0 em intervalos de 0.5. Os resultados podem ser observados na Tabela 5. Nesses experimentos, utilizamos o modelo FastText da Facebook Research, assim como o método SMOTE para balanceamento dos dados.

Limiar	Métricas		
	Prec	Rec	F1
1.5	0.09	0.95	0.17
2.0	0.17	0.89	0.29
2.5	0.25	0.63	0.35
3.0	0.31	0.57	0.40
3.5	0.31	0.54	0.40
4.0	0.35	0.54	0.42
4.5	0.36	0.54	0.43
5.0	0.37	0.54	**0.44**

Tabela 5: Resultados da avaliação da remoção de exemplos ruidosos

Dos resultados apresentados na Tabela 5, concluímos que a presença de pares de sentenças ruidosas parece possuir um efeito positivo no treinamento dos classificadores. Um explicação para esse fenômeno pode ser o fato desses exemplos servirem para informar os classificadores que enquanto a similaridade semântica textual parece ser uma importante evidência de paráfrase, o fenômeno de paráfrase não se limita o fenômeno de similaridade. Assim, tais sentenças informam ao classificador a existência de pares de sentença com alto grau de similaridade, mas que não são sentenças parafrásticas.

Por fim, calibrados os parâmetros experimentais, reavaliamos o nosso método sobre o corpus de teste, dessa vez usando o modelo FastText da Facebook Research, a representação pelo centroide, o classificador Naïve Bayes com balanceamento de dados usando a técnica SMOTE e sem exclusão de dados possivelmente ruidosos do conjunto de treino. Obtivemos então os seguintes resultados apresentados na Tabela 6.

Métricas		
Prec	Rec	F1
0.25	0.38	**0.30**

Tabela 6: Resultados da avaliação do método após calibração dos parâmetros experimentais

5 Considerações Finais

Este trabalho investigou a aplicação de diferentes métodos de representação de sentenças em um modelo de espaço vetorial da linguagem para o problema de identificação de paráfrase na língua portuguesa. Embora os resultados obtidos para a classificação de paráfrase tenham sido insatisfatórios, em comparação com os resultados relatados na literatura, acreditamos que nossos resultados indicam interessantes caminhos de investigação para detecção de paráfrases para a língua portuguesa. Particularmente, métodos simples de representação de sentenças e classificação, nomeadamente, a representação pelo centroide ou por semelhanças semânticas e um classificador Naïve Bayes, obtiveram os melhores resultados, indicando que uma grande quantidade de informações semânticas das sentenças são codificadas na geometria dos modelos de representação de palavras.

É importante notar que o desempenho das técnicas investigadas neste trabalho está claramente abaixo do desempenho relatado para o idioma inglês (c.f. (Kiros et al., 2015), por exemplo) ou para a inferência textual relatada pelos concorrentes no desafio ASSIN (c.f. (Barbosa et al., 2016) ou (Fialho et al., 2016)). Enquanto diversos parâmetros experimentais foram testados por nós, inúmeros aspectos do nosso projeto experimental devem ser considerados.

Primeiramente, por limitação dos recursos disponíveis para realização de experimentos, nós não pudemos avaliar a performance dos nossos prótipos quando treinados com os modelos com maior dimensionalidade treinados por Hartmann et al. (2017) – que obtiveram melhores resultados na avaliação por analogias. Enquanto nossa avaliação de parâmetros experimentais concluiu que o modelo de *word embedding* utilizado não foi fator determinante para os resultados, não pudemos avaliar o impacto da dimensionalidade de tais modelos – e portanto o poder de representação dos mesmos – nos resultados.

Note que o uso de técnicas de balanceamento de dados mais avançadas que a amostragem aleatória, utilizada nos experimentos originais, parecem ter efeito positivo na performance dos classificadores. Nesse sentido, resta realizar uma avaliação mais sistemática do efeito de técnicas de pré-processamento do conjunto de dados sobre a performance dos classificadores. Um importante ponto nesse sentido é a escolha de carac-

terísticas (*features*) para a descrição dos exemplos. A escolha feita por nós de utilizar duas formas de representação da relação semântica entre as sentenças presentes na literatura, tanto usando os vetores $\overrightarrow{u}$ e $\overrightarrow{v}$, quanto os vetores $\overrightarrow{u} \cdot \overrightarrow{v}$ e $\overrightarrow{u} - \overrightarrow{v}$, pode ter ocasionado um crescimento do espaço de representação, o que pode ter prejudicado o aprendizado dos classificadores.

Com relação ao desempenho dos métodos de representação pela agregação ponderada e SIF, notamos que apenas cerca de 393046 *tokens* no vocabulário do modelo FastText (composto por 592108 *tokens*) estão no dicionário IDF. Isso significa que cerca de 199062 *tokens* no modelo têm valor IDF de 0 e, portanto, não têm efeito na representação da sentença. Isso destaca que as diferentes estratégias de tokenização adotadas em nosso trabalho e a criação do modelo de *word embeddings* podem ter impactado nas representações que alcançamos e, portanto, nos resultados obtidos. É também digno de nota que o desempenho do método Skip-Thought pode ter sofrido com o fato de o corpus de treinamento ser relativamente pequeno em comparação com o utilizado para o idioma inglês (composto de 74.004.228 sentenças e 984.846.357 *tokens*).

É interessante observar que os métodos de melhor desempenho em nossos experimentos foram baseados na representação pelo centroide e pelas medidas de semelhança semântica entre as sentenças codificadas. Isso significa que a estrutura algébrica do espaço vetorial pode, na verdade, codificar uma grande quantidade de informações sobre a semântica composicional de sentenças e que um modelo simples de representação de sentenças pode ser adequado para muitas aplicações posteriores. Essas conexões teóricas e empíricas de *word embeddings* e semântica composicional, bem como as limitações do modelo codificador-decodificador, foram discutidas anteriormente na literatura, notadamente por Arora et al. (2018a,b); Dasgupta et al. (2018).

Note também que nossos experimentos utilizaram unicamente dados das representações vetoriais das sentenças e das similaridades obtidas através delas para treinar os classificadores. Outras características importantes e comumente utilizadas em sistemas de detecção de paráfrase e implicação textual, como medidas de sobreposição lexical, similaridades sintáticas, etc., poderiam ser trivialmente incorporados nos nossos modelos para melhorar a performance dos classificadores. Escolhemos, entretanto, não incorporar tais características em nosso modelo para avaliar quanta informação sobre o conteúdo semântico da sentença pode ser codificada na representação vetorial dessas sentenças.

Por fim, é importante salientar que o corpus ASSIN é constituído de exemplos difíceis, como evidenciado pelos resultados, assim como aqueles obtidos no trabalho de Gamallo e Pereira-Fariña (nesse volume) que utilizam o mesmo corpus para o problema de identificação de similaridade textual. Para avaliar o impacto da estrutura do ASSIN nos nossos resultados, pretendemos, no futuro, testar nossos métodos sobre o conjunto SICK-BR (Real et al., 2018) de inferência textual.

Referências

Alpaydin, Ethem. 2009. *Introduction to machine learning*. MIT Press.

Alves, Ana, Hugo Gonçalo Oliveira, Ricardo Rodrigues & Rui Encarnação. 2018. ASAPP 2.0: Advancing the state-of-the-art of semantic textual similarity for portuguese. Em *7th Symposium on Languages, Applications and Technologies (SLATE)*, 12:1–12:17.

Arora, Sanjeev, Yuanzhi Li, Yingyu Liang, Tengyu Ma & Andrej Risteski. 2018a. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association of Computational Linguistics* 6. 483–495.

Arora, Sanjeev, Yingyu Liang & Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. Em *5th International Conference on Learning Representations*, s.pp.

Arora, Sanjeev, Andrej Risteski & Yi Zhang. 2018b. Do GANs learn the distribution? some theory and empirics. Em *6th International Conference on Learning Representations*, s.pp.

Bahdanau, Dzmitry, Kyunghyun Cho & Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *Computing Research Repository* http://arxiv.org/abs/1409.0473.

Baptista, Jorge. 2018. Paraphrasing portuguese adverbs ending in –mente. Apresentado no POP - Por Outras Palavras. 1st Workshop on Linguistic Tools and Resources for Paraphrasing in Portuguese.

Barbosa, Luciano, Paulo Cavalin, Victor Guimaraes & Matthias Kormaksson. 2016. Blue man group no ASSIN: Usando representações distribuídas para similaridade semântica e inferência textual. *Linguamática* 8(2). 15–22.

de Barcelos Silva, Allan & Sandro José Rigo. 2018. Enhancing brazilian portuguese textual entailment recognition with a hybrid approach. *Journal of Computer Science* 14(7). 945–956. doi:10.3844/jcssp.2018.945.956.

Bojanowski, Piotr, Edouard Grave, Armand Joulin & Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606* .

Bruckschen, Mírian, Fernando Muniz, José Guilherme C. de Souza, Juliana Thiesen Fuchs, Kleber Infante, Marcelo Muniz, Patrícia Nunes Gonçalves, Renata Vieira & Sandra Aluísio. 2008. Anotação lingüística em XML do corpus PLN-BR. Relatório técnico. Universidade de São Paulo.

Cer, Daniel, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope & Ray Kurzweil. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175* .

Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall & W. Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research* 16. 321–357.

Conneau, Alexis, Douwe Kiela, Holger Schwenk, Loïc Barrault & Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. Em *Empirical Methods in Natural Language Processing*, 670–680.

Conneau, Alexis, Germán Kruszewski, Guillaume Lample, Loïc Barrault & Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *Computing Research Repository* http://arxiv.org/abs/1805.01070.

Cordeiro, João, Gaël Dias & Pavel Brázdil. 2007. A metric for paraphrase detection. Em *International Multi-Conference on Computing in the Global Information Technology (ICCGI)*, 35–40.

Dasgupta, Ishita, Demi Guo, Andreas Stuhlmüller, Samuel J. Gershman & Noah D. Goodman. 2018. Evaluating compositionality in sentence embeddings. *Computing Research Repository* http://arxiv.org/abs/1802.04302.

Dolan, Bill, Chris Quirk & Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. Em *5th International Conference on Intelligent Text Processing and Computational Linguistics*, s.pp.

Feitosa, David & Vladia Pinheiro. 2017. Análise de medidas de similaridade semântica na tarefa de reconhecimento de implicação textual. Em *11th Brazilian Symposium in Information and Human Language Technology*, 161–170.

Fernando, Samuel & Mark Stevenson. 2008. A semantic similarity approach to paraphrase detection. Em *11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, 45–52.

Fialho, Pedro, Ricardo Marques, Bruno Martins, Luísa Coheur & Paulo Quaresma. 2016. INESC-ID@ ASSIN: Medição de similaridade semântica e reconhecimento de inferência textual. *Linguamática* 8(2). 33–42.

Fonseca, Erick & Sandra M. Aluísio. 2018. Syntactic knowledge for natural language inference in portuguese. Em *International Conference on Computational Processing of the Portuguese Language*, 242–252. Springer.

Fonseca, Erick Rocha, Leandro Borges dos Santos, Marcelo Criscuolo & Sandra Maria Aluísio. 2016. Visão geral da avaliação de similaridade semântica e inferência textual. *Linguamática* 8(2). 3–13.

Gonçalo Oliveira, Hugo, Ana Oliveira Alves & Ricardo Rodrigues. 2017. Gradually improving the computation of semantic textual similarity in portuguese. Em *Progress in Artificial Intelligence*, 841–854.

Hartmann, Nathan, Erick R. Fonseca, Christopher Shulby, Marcos Vinícius Treviso, Jessica Rodrigues & Sandra M. Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *Computing Research Repository* http://arxiv.org/abs/1708.06025.

Hartmann, Nathan Siegle. 2016. Solo queue at ASSIN: Combinando abordagens tradicionais e emergentes. *Linguamática* 8(2). 59–64.

He, Haibo, Yang Bai, Edwardo A Garcia & Shutao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. Em *International Joint Conference on Neural Networks*, 1322–1328.

Howard, Jeremy & Sebastian Ruder. 2018. Fine-tuned language models for text classification. *Computing Research Repository* http://arxiv.org/abs/1801.06146.

Jing, Hongyan & Kathleen R. McKeown. 2000. Cut and paste based text summarization. Em *1st Annual Conference of the North American Chapter of the ACL*, 178–185.

Kenter, Tom & Maarten De Rijke. 2015. Short text similarity with word embeddings. Em *24th ACM International Conference on Information and Knowledge Management*, 1411–1420.

Kiros, Ryan, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun & Sanja Fidler. 2015. Skip-thought vectors. *Computing Research Repository* http://arxiv.org/abs/1506.06726.

Le, Quoc & Tomas Mikolov. 2014. Distributed representations of sentences and documents. Em *31st International Conference on Machine Learning*, 1188–1196.

Logeswaran, Lajanugen & Honglak Lee. 2018. An efficient framework for learning sentence representations. *arXiv preprint arXiv:1803.02893* s.pp.

Marelli, Marco, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini & Roberto Zamparelli. 2014. Semeval task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. Em *8th International Workshop on Semantic Evaluation*, 1–8.

Marsi, Erwin & Emiel Krahmer. 2005. Explorations in sentence fusion. Em *Tenth European Workshop on Natural Language Generation (ENLG)*, 109–117.

Mihalcea, Rada, Courtney Corley & Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. Em *21st National Conference on Artificial Intelligence*, 775–780.

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*, 3111–3119.

Miller, George A. 1995. WordNet: a lexical database for English. *Communications of the ACM* 38(11). 39–41.

Patro, Badri N., Vinod K. Kurmi, Sandeep Kumar & Vinay P. Namboodiri. 2018. Learning semantic sentence embeddings using pair-wise discriminator. *arXiv preprint arXiv:1806.00807* s.pp.

Pennington, Jeffrey, Richard Socher & Christopher Manning. 2014. Glove: Global vectors for word representation. Em *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Pinheiro, Anderson, Rafael Ferreira, Máverick Dionísio, Vitor Rolim & oão Tenório. 2017. Statistical and semantic features to measure sentence similarity in portuguese. Em *Brazilian Conference on Intelligent Systems (BRACIS)*, 342–347. doi:10.1109/BRACIS.2017.40.

Real, Livy, Ana Rodrigues, Andressa Vieira e Silva, Beatriz Albiero, Bruna Thalenberg, Bruno Guide, Cindy Silva, Guilherme de Oliveira Lima, Igor Câmara, Miloš Stanojević et al. 2018. SICK-BR: a Portuguese corpus for inference. Em *13th International Conference on Computational Processing of the Portuguese Language*, 303–312.

Rocha, Gil & Henrique Lopes Cardoso. 2018. Recognizing textual entailment: Challenges in the Portuguese language. *Information* 9(4). 76.

Sahlgren, Magnus. 2008. The distributional hypothesis. *Italian Journal of Disability Studies* 20. 33–53.

Shinyama, Yusuke, Satoshi Sekine & Kiyoshi Sudo. 2002. Automatic paraphrase acquisition from news articles. Em *2nd International Conference on Human Language Technology Research*, 313–318.

Silva, Allan, Sandro Rigo, Isa Mara Alves & Jorge Barbosa. 2017. Avaliando a similaridade semântica entre frases curtas através de uma abordagem híbrida. Em *11th Brazilian Symposium in Information and Human Language Technology*, 93–102.

Socher, Richard, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng & Christopher D. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. Em *Advances in Neural Information Processing Systems*, 801–809.

Souza, Marlo & Leandro M. P. Sanches. 2018. Detecting paraphrases for Portuguese using word and sentence embeddings. Apresentado no POP - Por Outras Palavras. 1st Workshop on Linguistic Tools and Resources for Paraphrasing in Portuguese.

Suresh, Subhashree & P. Sreenivasa Kumar. 2016. Enriching linked datasets with new object properties. *Computing Research Repository* http://arxiv.org/abs/1606.07572.

Turney, Peter D. & Michael L. Littman. 2002. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. *Computing Research Repository* http://arxiv.org/abs/cs.LG/0212012.

Yang, Yinfei, Steve Yuan, Daniel Cer, Sheng-yi Kong, Noah Constant, Petr Pilar, Heming Ge, Yun-Hsuan Sung, Brian Strope & Ray Kurzweil. 2018. Learning semantic textual similarity from conversations. *Computing Research Repository* http://arxiv.org/abs/1804.07754.

Yin, Wenpeng & Hinrich Schütze. 2015. Convolutional neural network for paraphrase identification. Em *Annual Conference of the North American Chapter of the ACL: Human Language Technologies*, 901–911.