Proposta recebida em Setembro 2016 e aceite para publicação em Novembro 2016.

ASAPP: Alinhamento Semântico Automático de Palavras aplicado ao Português

ASAPP: Automatic Semantic Alignment for Phrases applied to Portuguese

Ana Oliveira Alves
CISUC, Universidade de Coimbra
ISEC, Instituto Politécnico de Coimbra
ana@dei.uc.pt

Ricardo Rodrigues
CISUC, Universidade de Coimbra
ESEC, Instituto Politécnico de Coimbra
rmanuel@dei.uc.pt

Hugo Gonçalo Oliveira
CISUC, Universidade de Coimbra
DEI, Universidade de Coimbra
hroliv@dei.uc.pt

Resumo

Apresentamos duas abordagens distintas à tarefa de avaliação conjunta ASSIN onde, dada uma coleção de pares de frases escritas em português, são colocados dois objectivos para cada par: (a) calcular a similaridade semântica entre as duas frases; e (b) verificar se uma frase do par é paráfrase ou inferência da outra. Uma primeira abordagem, apelidada de Reciclagem, baseia-se exclusivamente em heurísticas sobre redes semânticas para a língua portuguesa. A segunda abordagem, apelidada de ASAPP, baseia-se em aprendizagem automática supervisionada. Acima de tudo, os resultados da abordagem Reciclagem permitem comparar, de forma indireta, um conjunto de redes semânticas, através do seu desempenho nesta tarefa. Estes resultados, algo modestos, foram depois utilizados como características da abordagem ASAPP, juntamente com características adicionais, ao nível lexical e sintático. Após comparação com os resultados da coleção dourada, verifica-se que a abordagem ASAPP supera a abordagem Reciclagem de forma consistente. Isto ocorre tanto para o Português Europeu como para o Português Brasileiro, onde o desempenho atinge uma exatidão de 80.28%±0.019 para a inferência textual, enquanto que a correlação dos valores atribuídos para a similaridade semântica com aqueles atribuídos por humanos é de 66.5% ± 0.021.

Palavras chave

similaridade semântica, inferência textual, redes léxico-semânticas, aprendizagem automática

Abstract

We present two distinct approaches to the ASSIN shared evaluation task where, given a collection with pairs of sentences, in Portuguese, poses the following challenges: (a) computing the semantic similarity between the sentences of each pair; and (b) testing whether one sentence paraphrases or entails the other. The first approach, dubbed Reciclagem, is exclusively based on heuristics computed on Portuguese semantic networks. The second, dubbed ASAPP, is based on supervised machine learning. The results of Reciclagem enable an indirect comparison of Portuguese semantic networks. They were then used as features of the ASAPP approach, together with lexical and syntactic features. After comparing our results with those in the gold collection, it is clear that ASAPP consistently outperforms Reciclagem. This happens both for European Portuguese and Brazilian Portuguese, where the entailment performance reaches an accuracy of 80.28% ± 0.019, and the semantic similarity scores are 66.5% ± 0.021 correlated with those given by humans.

Keywords

semantic similarity, entailment, lexical semantic networks, machine learning

1 Introdução

A Similaridade Semântica e Inferência Textual (em inglês, *Entailment*) têm sido alvo de intensa pesquisa por parte da comunidade científica em Processamento da Linguagem Natural. Prova disso é a organização de várias tarefas de avaliação sobre o tema (*Semantic Textual Similarity — STS*) e o surgimento de conjuntos de dados anotados nos últimos anos[1] (Agirre et al., 2015,

[1] Veja-se, por exemplo, a tarefa mais recente, SemEval-2016 STS Task: http://alt.qcri.org/semeval 2016/task1/

2014, 2013, 2012). No capítulo 2 deste artigo, são precisamente apresentados trabalhos que têm o objectivo comum de calcular a similaridade e inferência textual, assim como tarefas que incentivem esta pesquisa.

No entanto, as tarefas anteriores, realizadas no âmbito das avaliações SemEval, focavam apenas a língua inglesa. A tarefa ASSIN, em que nos propusemos participar, tem algumas semelhanças com as anteriores, mas visa a língua portuguesa. Dada uma coleção com pares de frases, o objectivo dos sistemas participantes passa por: (a) atribuir um valor para a similaridade de cada par; e (b) classificar cada par como paráfrase, inferência, ou nenhum dos anteriores.

A nossa participação na tarefa ASSIN seguiu dois caminhos distintos e, consequentemente, duas equipas participantes, ainda que constituídas pelos mesmos elementos, e onde foram utilizados os mesmos recursos e ferramentas para o processamento computacional da língua (estes são apresentados no capítulo 3). A primeira abordagem – Reciclagem – baseou-se exclusivamente no cálculo de heurísticas sobre um conjunto de redes em que palavras portuguesas estão organizadas de acordo com os seus possíveis sentidos.

A segunda abordagem tem como inspiração o sistema ASAP – *Automatic Semantic Alignment for Phrases* – que, numa primeira versão, participou na tarefa de *Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Textual Entailment* do SemEval 2014 (Alves et al., 2014) e, numa segunda instanciação, na tarefa de Semantic Textual Similarity do SemEval 2015 (Alves et al., 2015). O nome do sistema aqui apresentado acrescenta um *P* ao nome do sistema original, por se focar na língua portuguesa.

Tanto o ASAP como o ASAPP vêem a Similaridade Textual e o *Entailment* como uma função onde as variáveis são as características lexicais, sintáticas e semânticas extraídas do texto. A extração destas características nas suas várias dimensões é detalhada no capítulo 4. Uma das nossas principais contribuições prende-se com a possibilidade de comparar uma abordagem heurística com uma abordagem aprendida de forma supervisionada pela máquina (capítulo 6) para um mesmo conjunto de características na língua Portuguesa, seja na variante Europeia ou na Brasileira. Há a referir que os resultados das heurísticas de similaridade calculadas na abordagem Reciclagem são também utilizados como características da abordagem ASAPP.

Várias ferramentas foram utilizadas para a extração das características morfo-sintáticas. Estas incluem a atomização (em inglês, *tokenization*), etiquetagem gramatical (*part-of-speech tagging*), lematização, segmentação de orações (*chunking*) e reconhecimento de entidades mencionadas, que são explicadas em detalhe na secção 3.1. Quanto às características semânticas, um conjunto de redes léxico-semânticas foi explorado e é introduzido na secção 3.2. Nestas redes, que pretendem ter uma boa cobertura da língua portuguesa, as palavras encontram-se organizadas de acordo com os seus sentidos. Elas são utilizadas para identificar relações entre palavras das duas frases do par.

Os resultados de ambas abordagens seguindo diversas combinações de características e aplicação de diferentes algoritmos de aprendizagem são discutidos no capítulo 7. Por fim, o capítulo 8 reúne as principais conclusões que foram determinadas a partir destes resultados e sua discussão.

2 Trabalho Relacionado

Existem atualmente duas abordagens principais para o cálculo da similaridade. A primeira consiste no uso de um corpo de grande dimensão para estimar a similaridade através de dados estatísticos recolhidos sobre a co-ocorrência de palavras. A segunda é baseada em conhecimento léxico-semântico, utilizando relações e entradas de um dicionário (Lesk, 1986) ou recurso léxico-semântico (Banerjee & Pedersen, 2003). As abordagem híbridas combinam as duas metodologias (Jiang & Conrath, 1997).

O algoritmo de Lesk (Lesk, 1986) utiliza definições de entradas de um dicionário (sentidos) para desambiguar uma palavra polissémica no contexto de uma frase. O principal objectivo deste método é contar o número de palavras que são comuns entre duas definições, no caso do cálculo da similaridade entre duas entradas do dicionário. Em alguns casos, as definições obtidas são muito reduzidas em tamanho e mostram-se insuficientes para identificar similaridades entre sentidos relacionados de palavras. Para aperfeiçoar este método, Banerjee & Pedersen (2003) adaptaram o algoritmo para utilizar a base de conhecimento léxico-semântico WordNet (Fellbaum, 1998) como dicionário, onde é possível encontrar as definições dos sentidos das palavras, e estenderam a medida de Lesk para a utilização da rede de relações semânticas entre conceitos, na WordNet.

A métrica de similaridade de Jiang & Conrath (1997) calcula a informação partilhada entre conceitos, que é determinada pelo Conteúdo da Informação (*Information Content – IC*) do conceito mais específico que seja o hiperónimo de dois conceitos que se pretende comparar. Utilizando a hierarquia de hiperónimos/hipónimos da WordNet, esta medida calcula a distância (inverso da similaridade) entre dois conceitos, através da contagem de relações deste tipo.

Mais recentemente, a tarefa de Similadade Semântica e Inferência Textual para o inglês têm ocorrido desde 2012 nos workshops internacionais de avaliação semântica (Semeval-STS), providenciando um fórum privilegiado para a avaliação de algoritmos e modelos. Na última tarefa realizada, dos sistemas participantes, o vencedor foi uma abordagem baseada em técnicas de *deep learning* com sinais de penalização e reforço aplicados à rede recorrente extraídos do WordNet(Rychalska et al., 2016) que podem ser combinadas em conjuntos (*ensemble*) de classificadores. Os autores incluíram ainda neste conjunto uma versão do algoritmo do ano anterior (Sultan et al., 2015) melhorado através do uso de características que incluem *word embbedings*.

Os métodos de reconhecimento de inferência textual baseiam-se geralmente na assunção que duas expressões em linguagem natural podem ser inferidas uma a partir de outra. A paráfrase é um caso especial de inferência textual bidireccional, onde estas duas expressões transmitem de uma forma muito aproximada a mesma informação. Existem diferentes abordagens para identificar a inferência textual (Androutsopoulos & Malakasiotis, 2010), baseadas em: lógica computacional; similaridade lexical de palavras presentes nos pares de expressões; similaridade sintática das expressões; construção de um mapeamento semântico entre os pares de expressão, de acordo com um modelo vectorial.

Dada a inexistência de coleções de teste para este tipo de tarefas, os trabalhos focados na língua portuguesa são escassos. Seno & Nunes (2008) identificam e agrupam frases semelhantes numa coleção de documentos escritos em Português do Brasil. A distância entre pares de frases é calculada com base no número de palavras em comum, e em duas métricas: o TF-IDF (frequência de um termo multiplicada pela sua frequência inversa nos documentos da coleção) e o TF-ISF (frequência de um termo multiplicada pela sua frequência inversa nas frases da coleção).

Mais recentemente, Pinheiro et al. (2014) apresentaram uma abordagem precisamente à tarefa de STS para português, baseada nos conteúdos da base de conhecimento InferenceNet.Br, utilizada para identificar palavras relacionadas em duas frases comparadas. A medida proposta foi avaliada numa coleção com a descrição de erros reportados num conjunto de projetos de engenharia de software, cuja similaridade foi posteriormente anotada por dois juízes humanos. O objetivo seria recuperar erros semelhantes.

Relativamente à inferência textual, Barreiro (2008) estudou o parafraseamento de frases portuguesas com base em verbos de suporte e analisou o impacto da realização destas paráfrases na tradução automática das frases para inglês.

3 Ferramentas e Recursos PLN

Apresentamos aqui o conjunto de ferramentas e recursos base utilizado neste trabalho para o processamento computacional da língua portuguesa. Mais propriamente, enumeram-se as ferramentas utilizadas para a anotação morfo-sintática das frases e, de seguida, as redes de onde foram obtidas as características semânticas.

3.1 Anotação Morfo-Sintática

Diversas ferramentas foram utilizadas para o processamento das frases da coleção ASSIN, nomeadamente um atomizador (em inglês, *tokenizer*), um etiquetador gramatical (*part-of-speech tagger*), um lematizador – tanto na nossa abordagem heurística como na supervisionada – e ainda um reconhecedor de entidades mencionadas e um segmentador de orações ("*phrase chunker*") – utilizados exclusivamente pela abordagem ASAPP.

À exceção do lematizador, todas as ferramentas para anotação morfo-sintática tiveram como base o Apache OpenNLP Toolkit[2], utilizando modelos de máxima entropia, com algumas alterações que identificamos nas descrições que se seguem.

3.1.1 Atomização

A tarefa de atomização tem como objetivo separar as frases em átomos simples. Para esta tarefa, foi usado como ponto de partida o *tokenizer* do OpenNLP com o modelo para o português[3], com o resultado a ser alvo de pós-processamento, com vista a melhorar a sua qualidade. Por exemplo, o resultado inicial é analisado para a eventual identificação da presença de clíticos, procurando

[2] http://opennlp.apache.org/
[3] http://opennlp.sourceforge.net/models-1.5/

separar formas verbais de pronomes átonos, de forma a melhorar posteriormente o desempenho do etiquetador gramatical (e.g., *dar-me-ia* → *daria a mim*). O mesmo acontece com as contrações, de forma a separar preposições de pronomes ou determinantes (e.g., *ao* → *a o*). Para além dos clíticos e das contrações, também as abreviações são alvo de análise: na prática, para reverter eventuais casos em que abreviações compostas possam ter sido separadas nos resultados iniciais do *tokenizer* (e.g., *q. b.* → *q.b.*).

3.1.2 Etiquetagem Gramatical

Para a etiquetagem gramatical, foi também utilizado o Apache OpenNLP. Neste caso, dados os cuidados anteriores com a atomização, cujos resultados são usados como entrada do etiquetador, verificou-se que a utilização do modelo já disponibilizado também pelo OpenNLP seria suficiente. Ou seja, os resultados obtidos com o *PoS tagger* do OpenNLP foram utilizados diretamente nos restantes passos, salvo pequenos aspetos para melhor integração na restante abordagem. As possíveis etiquetas gramaticais são adjetivo, advérbio, artigo, nome, numeral, nome próprio, preposição e verbo. Se assim desejarmos, também a pontuação pode ser anotada.

3.1.3 Lematização

Para a lematização dos termos presentes nas frases, foi utilizado o LemPORT (Rodrigues et al., 2014), um lematizador baseado em regras e também na utilização de um léxico constituído pelas formas base dos termos e respetivas declinações.

Recebendo como entrada termos (*átomos*) e respetivas etiquetas gramaticais, o LemPORT começa por utilizar o léxico e, dando-se o caso de o termo a lematizar já existir no léxico, devolve a forma base correspondente. Contudo, sendo um léxico um recurso que, por natureza da própria língua, não pode compreender todas as palavras existentes ou usadas, são utilizadas regras para normalizar os termos não incluídos, em função do modo, número, grau (superlativo, aumentativo e diminutivo), género e conjugações, aplicando-se, consoante os casos, a cada uma das categorias gramaticais, mas com maior peso em substantivos, adjetivos e verbos. Neste caso, o léxico é novamente utilizado para validar o resultado da aplicação das regras – regra após regra, determinando quando parar a sua execução. Quando o resultado continua a não constar do léxico, é usado como critério de término a exaustão das regras aplicáveis.

3.1.4 Reconhecimento de EM

Para o reconhecimento de entidades mencionadas (REM) – aqui enquadrado, apesar de as entidades serem, na verdade, uma característica semântica – voltou a ser utilizado o Apache OpenNLP, aqui com a diferença de não existir um modelo já criado para o efeito. Foi assim necessário criar um modelo que se baseou no corpo Amazónia[4], um dos corpos que compõem a "Floresta Sintá(c)tica" (Afonso et al., 2001), disponibilizado pela Linguateca[5]. Este corpo é composto por cerca de 4,6 milhões de palavras, correspondentes a cerca de 275 mil frases, retiradas de uma plataforma colaborativa *on-line* referente à produção cultural brasileira, recolhidas em Setembro de 2008 (Freitas & Santos, 2015). O corpo foi utilizado tanto para treinar como para testar o modelo, tendo-se alcançado uma precisão de 0,80, uma abrangência de 0,75, e uma medida $F1$ de 0,77[6]. Quanto aos resultados do REM, estes foram utilizados diretamente (tal como apresentados pelo *entity finder* do OpenNLP), também salvos pequenos aspetos para melhor integração na restante abordagem. Relativamente aos diversos tipos de entidade mencionada identificados, estes são: abstrações, artigos & produtos, eventos, números, organizações, pessoas, lugares, coisas e datas & horas. Importa também referir que os termos identificados pelo *tokenizer* são usados como entrada no reconhecedor de entidades mencionadas.

3.1.5 Segmentação de Orações

Para a segmentação de orações, de forma semelhante ao que aconteceu com o REM, foi utilizado o Apache OpenNLP, tendo ainda havido necessidade de criar um modelo para o efeito. Neste caso, foi utilizado o Bosque 8.0, outro dos corpos constituintes da "Floresta Sintá(c)tica", mais uma vez para treinar e para testar o modelo, tendo-se alcançado uma precisão de 0,95, uma abrangência de 0,96, e medida $F1$ de 0,95. O segmentador tem como entrada os "tokens" e as respetivas etiquetas gramaticais, bem como os lemas. As orações podem ser classificadas como nominais, verbais ou preposicionais. Novamente,

[4] http://www.linguateca.pt/floresta/corpus.html
[5] http://www.linguateca.pt/
[6] Relativamente aos valores de precisão, abrangência e $F1$, da ferramenta e modelo de REM utilizados, interessa reforçar que foram obtidos usando também o corpo Amazónia (80% para treino e 20% para teste). Usando o mesmo corpo para treino, mas outro para teste (a coleção dourada do HAREM (Mota, 2007)), Fonseca et al. (2015) encontraram valores bastantes distintos, com 37,97% para precisão, 38,14% para abrangência e 38,06% para $F1$.

à exceção de pequenos aspetos relacionados com a apresentação dos resultados, incluindo-se na descrição das orações também os lemas (que não são considerados na versão original do *chunker* OpenNLP), estes foram utilizados diretamente na abordagem.

3.2 Redes Semânticas

O conhecimento sobre as palavras de uma língua e os seus possíveis sentidos pode organizar-se nas chamadas bases de conhecimento léxico-semântico onde, para o inglês, se destaca a WordNet de Princeton (Fellbaum, 1998). Entre as várias tarefas do processamento computacional da língua que podem recorrer a uma destas bases de conhecimento, destaca-se a similaridade semântica.

Para o português, existem atualmente vários recursos computacionais com características semelhantes à WordNet, inclusivamente várias wordnets (Gonçalo Oliveira et al., 2015). Alternativamente a escolher uma base de conhecimento, neste trabalho foram utilizados vários recursos desse tipo, todos eles abertos. Testaram-se várias métricas para o cálculo da similaridade semântica com base em cada um dos recursos e algumas combinações. De certa forma, podemos ver esta parte do trabalho como uma comparação indireta dos recursos nas tarefas alvo. Mais propriamente, foram utilizadas redes semânticas $R(P, L)$, com $|N|$ palavras (nós) e $|L|$ ligações entre palavras. Cada ligação tem associado o nome de uma relação semântica (e.g. SINÓNIMO-DE, HIPERÓNIMO-DE, PARTE-DE, ...) e define um triplo $palavra_1$ *relacionada-com* $palavra_2$ (e.g. *animal* HIPERÓNIMO-DE *cão*, *roda* PARTE-DE *carro*). As redes utilizadas foram obtidas a partir dos seguintes recursos:

- PAPEL (Gonçalo Oliveira et al., 2008), relações extraídas automaticamente a partir do Dicionário da Língua Portuguesa da Porto Editora, com recurso a gramáticas baseadas nas regularidades das definições;
- Dicionário Aberto (Simões et al., 2012) e Wikcionário.PT [7], dois dicionários de onde foram extraídas relações com base nas mesmas gramáticas que no PAPEL, e integrados na rede CARTÃO (Gonçalo Oliveira et al., 2011);
- TeP 2.0 (Maziero et al., 2008) e OpenThesaurus.PT[8], dois tesauros que agrupam palavras com os seus sinónimos, no que vulgarmente se chama de *synset*;
- OpenWordNet-PT (OWN.PT) (de Paiva et al., 2012) e PULO (Simões & Guinovart, 2014), duas wordnets.

Dos recursos anteriores, aqueles que não se encontram disponíveis no formato referido anteriormente foram nele convertidos. Assim, para os tesauros e para as wordnets, cada par de palavras agrupado num *synset* deu origem a uma relação de sinonímia. Para as wordnets, foi ainda criada uma relação para cada par de palavras em dois *sysets* relacionados. Por exemplo, uma relação do tipo PARTE-DE entre os *synsets* {*porta, portão*} e {*automóvel, carro, viatura*} resultaria nos seguintes triplos: (*porta* SINÓNIMO-DE *portão*), (*automóvel* SINÓNIMO-DE *carro*), (*automóvel* SINÓNIMO-DE *viatura*), (*carro* SINÓNIMO-DE *viatura*), (*porta* PARTE-DE *automóvel*), (*porta* PARTE-DE *carro*), (*porta* PARTE-DE *viatura*), (*portão* PARTE-DE *automóvel*), (*portão* PARTE-DE *carro*), (*portão* PARTE-DE *viatura*).

Finalmente, foi também utilizada a versão mais recente do CONTO.PT (Gonçalo Oliveira, 2016), uma wordnet difusa baseada na redundância de informação nos recursos anteriores. Os *synsets* do CONTO.PT foram descobertos de forma automática, com base nas relações de sinonímia nos vários recursos, e incluem palavras com valores de pertença variáveis, indicadores de confiança – quanto maior esse valor, maior a confiança na utilização da palavra para transmitir o significado do *synset*. Inclui ainda um conjunto de valores de confiança associados a cada relação entre *synsets*.

4 Extração de características

As caraterísticas obtidas a partir de dados em bruto permitem que estes possam ser trabalhados por algoritmos heurísticos (baseados em conhecimento) ou de aprendizagem pela máquina. Quando se trata de processamento da linguagem natural escrita, estas características podem envolver as diversas fases de análise tais como: Lexical, Sintática, Semântica e do Discuso. Considerando que a coleção ASSIN é composta essencialmente por pares de frases isoladas, torna-se difícil ter um contexto mais amplo para análise do discurso. Sendo assim, foram consideradas as três primeiras análises para a extração de características. O nosso principal objetivo é extrair características de forma completamente automática, com base em ferramentas

[7] http://pt.wiktionary.org
[8] http://paginas.fe.up.pt/~arocha/AED1/0607/trabalhos/thesaurus.txt

e recursos existentes. Apesar de algumas características terem sido avaliadas de forma independente (capítulo 5), cada uma pode ser considerada uma métrica de similaridade parcial, parte de uma análise de regressão (capítulo 6).

4.1 Características Lexicais

Considerando as palavras presentes nos pares de frases da coleção ASSIN, foram contabilizadas:

- Contagem de palavras e expressões consideradas negativas[9] presentes em cada frase (Cn_{f1} e Cn_{f2}). Assim como o valor absoluto da diferença entre estas duas contagens ($|Cn_{f1} - Cn_{f2}|$), sempre calculadas após a lematização de cada palavra;
- Contagem dos átomos em comum nas duas frases;
- Contagem dos lemas em comum nas duas frases.

4.2 Características Morfo-Sintáticas

Tendo em consideração a estrutura das frases e utilizando o segmentador de orações apresentado na secção 3.1.5, foram contabilizadas as contagens de grupos nominais, verbais e preposicionais em cada uma das frases de cada par, e calculado o valor absoluto da diferença para cada tipo de grupo.

Ainda com as ferramentas introduzidas na secção 3.1, o REM foi aplicado de forma a identificar a presença de entidades mencionadas (EM) em cada uma das frases. Para cada tipo de EM[10] foi calculado o valor absoluto da diferença da contagem em ambas as frases de cada par da coleção ASSIN.

4.3 Características semânticas

As características semânticas foram calculadas com recurso às redes apresentadas na secção 3.2. Um primeiro conjunto de características baseou-se exclusivamente na contagem de palavras da primeira frase de cada par relacionadas com palavras da segunda frase respetiva.

Para além das contagens, foi calculada a similaridade semântica de cada par de frases, com base em heurísticas aplicadas sobre as redes semânticas. Algumas dessas heurísticas foram inspiradas em trabalhos relacionados, inclusivamente para o português e sobre algumas das mesmas redes semânticas (Gonçalo Oliveira et al., 2014).

As heurísticas aplicadas podem agrupar-se em três tipos:

- Semelhança entre as vizinhanças das palavras nas redes;
- Baseadas na estrutura das redes de palavras;
- Baseadas na presença e pertença em *synsets* difusos.

4.3.1 Semelhança entre as vizinhanças

O primeiro grupo de heurísticas inclui diferentes formas de calcular a semelhança entre conjuntos que, neste caso, são formados pela palavra alvo e por as que lhe são adjacentes na rede semântica, a que chamamos a vizinhança (viz, na equação 1).

$$\begin{aligned} Viz(palavra) = & sinonimos(palavra) \\ & \cup\, hiperonimos(palavra) \\ & \cup\, hiponimos(palavra) \\ & \cup\, partes(palavra) \\ & \cup\, ... \end{aligned} \tag{1}$$

O conjunto das palavras vizinhas podia incluir efetivamente todas as palavras diretamente relacionadas, ou poderia restringir-se apenas a alguns tipos de relação. Por exemplo, em algumas experiências utilizaram-se apenas sinónimos e hiperónimos.

Para calcular a similaridade entre duas frases, t e h, cada uma é representada como um conjunto de palavras, T e H. Partindo da vizinhança de cada palavra, a similaridade das frases é calculada de uma de três formas:

- Total: para cada par de frase é primeiro criado um conjunto, C_t e C_h, que reúne as vizinhanças de todas as palavras da frase t e h, respetivamente (equação 2)[11].

$$C_F = \bigcup_{i=1}^{|F|} Viz(F_i) \tag{2}$$

  Neste caso, a similaridade é igual à semelhança entre C_t e C_h (equação 3).

$$Sim_{Total}(t,h) = Sem(C_t, C_h) \tag{3}$$

[9] Palavras tais como: "não", "de modo algum", "de forma alguma", "coisa alguma", "nada", "nenhum", "nenhuma", "nem", "ninguém", "nunca", "jamais", "proibido", "sem", "contra", "incapaz."

[10] abstrações, artigos & produtos, eventos, números, organizações, pessoas, lugares, coisas e datas & horas.

[11] Podem ser consideradas efetivamente todas as palavras ou apenas aquelas com determinada categoria gramatical. Neste caso, foram apenas utilizadas palavras de categoria aberta, ou seja, substantivos, verbos, adjetivos e advérbios.

- $m \times n$: a similaridade é calculada com base na semelhança média entre a vizinhança de cada palavra de T com cada palavra de H (equação 4).

$$Sim_{n\times m}(t,h) = \sum_{i=1}^{|T|}\sum_{j=1}^{|H|} Sem(Viz(T_i), Viz(H_j)) \tag{4}$$

- $Max(m \times n)$: semelhante ao anterior mas, para cada palavra em T é apenas considerada a semelhança mais elevada com uma palavra de H.

$$Sim_{max}(t,h) = \sum_{i=1}^{|t|} max\Big(Sim\big(Viz(T_i), Viz(H_j)\big)\Big) : H_j \in H \tag{5}$$

Por sua vez, a semelhança entre as vizinhanças podia ser calculada com base em uma de quatro heurísticas, todas elas adaptações do algoritmo de Lesk (Banerjee & Pedersen, 2003). A semelhança entre duas palavras podia então ser dada pelo cardinal da intersecção das suas vizinhanças (equação 6), ou pelos coeficientes de Jaccard (equação 7), Overlap (equação 8) ou Dice (equação 9), também das suas vizinhanças.

$$Lesk(A,B) = |Viz(A) \cap Viz(B)| \tag{6}$$

$$Jaccard(A,B) = \frac{|Viz(A) \cap Viz(B)|}{|Viz(A) \cup Viz(B)|} \tag{7}$$

$$Overlap(A,B) = \frac{|Viz(A) \cap Viz(B)|}{min(|Viz(A)|, |Viz(B)|)} \tag{8}$$

$$Dice(A,B) = 2.\frac{|Viz(A) \cup Viz(B)|}{|Viz(A)| + |Viz(B)|} \tag{9}$$

Enquanto que os três coeficientes estão dentro do intervalo $[0,1]$, a intersecção está no intervalo $[0,+\infty]$. Foi por isso normalizada no intervalo $[0,1]$, através da divisão do cardinal da intersecção pelo valor da maior intersecção para as frases comparadas.

4.3.2 Heurísticas baseadas na estrutura da rede

Foram aplicadas duas medidas que exploram a estrutura da rede, nomeadamente:

- Distância média: entre cada par de palavras em que a primeira palavra é da frase t e a segunda é da frase h. Neste caso, a similaridade seria o inverso da distância média.
- *Personalized PageRank* (Agirre & Soroa, 2009): para se ordenarem os nós da rede de acordo com a sua relevância estrutural para cada frase f é feito o seguinte:
  1. Atribuição de um peso a cada nó da rede semântica, que será $\frac{1}{|F|}$, se o nó corresponder a uma palavra da frase f, ou 0, caso contrário;
  2. Com os pesos anteriores, execução do algoritmo de PageRank na rede;
  3. Ordenamento dos nós da rede de acordo com o seu peso após 30 iterações;
  4. Criação de um conjunto E_{fn} com as primeiras n palavras.

  A similaridade entre t e h é depois calculada através da intersecção entre E_{tn} e E_{hn}. Nas experiências realizadas, utilizou-se $n = 50$.

4.3.3 Heurística baseada na presença e pertença em synsets difusos

Para se utilizar a rede CONTO.PT, a abordagem foi um pouco diferente, também devido às diferentes características desta rede. A CONTO.PT é estruturada em *synsets* difusos, onde cada palavra tem um valor de pertença, para além de relações entre *synsets*, cada uma com um valor de confiança associado. Nesta heurística verifica-se se, para cada par de palavras, $(p1,p2) : p_1 \in h$ e $p_2 \in t$:

1. Há pelo menos um *synset* $S_{12} : p_1 \in S_{12} \wedge p_2 \in S_{12}$. Neste caso, a similaridade das palavras será igual à soma das suas pertenças nesse *synset*, multiplicada por um peso ρ_s. Matematicamente, $Sim(p_1,p_2) = (\mu(p_1,S_1) + \mu(p_2,S_2)) \times \rho_s$
2. Há pelo menos dois *synsets* $S_1, S_2 : p_1 \in S_1 \wedge p_2 \in S_2$ relacionados. Neste caso, a similaridade é igual à soma das suas pertenças em cada um desses *synsets*, multiplicada pela soma da confiança na relação e ainda por um peso, que será ρ_h para hiperonímia ou ρ_o para outro tipo de relação, em que fará sentido que $\rho_s > \rho_h > \rho_o$. Matematicamente, $Sim(p_1,p_2) = (\mu(p_1,S_1) + \mu(p_2,S_2)) \times conf(S_1, Relacao, S_2) \times \rho$

A similaridade das frases t e h resulta depois da soma da similaridade máxima entre cada palavra de t e qualquer outra palavra de h. Admitimos que este tipo de rede poderia ter sido mais explorado, o que acabou por não acontecer.

4.3.4 Contagens de Relações

Para além das heurísticas anteriores, um outro conjunto de características semânticas utilizadas

pelo sistema ASAPP baseou-se na contagem simples de relações entre palavras de uma e outra frase do par. Mais precisamente, para cada rede semântica, foram extraídas quatro contagens: (i) sinonímia; (ii) hiperonímia/hiponímia; (iii) antonímia; e (iv) outras relações.

A título de exemplo, considere-se o seguinte par de frases:

- *Além de Ishan, a polícia pediu ordens de detenção de outras 11 pessoas, a maioria deles estrangeiros.*
- *Além de Ishan, a polícia deu ordem de prisão para outras 11 pessoas, a maioria estrangeiros.*

Com base na rede PAPEL, as seguintes contagens são obtidas:

- $Sinonimia = 3$ — {(*polícia*, *ordem*), (*ordem*, *polícia*), (*detenção*, *prisão*)}
- $Hiponimia = 1$ — {(*estrangeiro*, *pessoa*)}
- $Antonimia = 0$
- $Outras = 2$ — {(*polícia* SERVE_PARA *ordem*), (*ordem* FAZ_SE_COM *polícia*)}

5 Reciclagem

Reciclagem é um sistema exclusivamente baseado em conhecimento léxico-semântico que procura calcular a similaridade de frases sem qualquer tipo de supervisão. Para tal, ele utiliza unicamente as heurísticas anteriormente apresentadas. Ou seja, dado um par de frases, uma rede semântica e uma heurística, ele calcula um valor para a similaridade das frases.

Apesar dos resultados destas heurísticas serem depois utilizados como características do sistema ASAPP, o sistema Reciclagem tem dois objetivos principais:

- Verificar até que ponto uma abordagem não supervisionada se equipara a uma abordagem que recorre a treino. Por exemplo, para o inglês, a exploração de bases de conhecimento léxico-semântico levou a resultados comparáveis aos de abordagens supervisionadas em tarefas como a desambiguação do sentido das palavras (Agirre et al., 2009; Ponzetto & Navigli, 2010).
- Realizar uma comparação indireta de um leque das bases de conhecimento léxico-semântico atualmente disponíveis para a língua portuguesa, através do seu desempenho no cálculo de similaridade semântica. Uma comparação noutra tarefa, mas com algumas semelhanças, foi apresentada em Gonçalo Oliveira et al. (2014).

O cálculo da similaridade é realizado após uma fase de pré-processamento, onde as frases são atomizadas e onde os átomos recebem anotações morfo-sintáticas, para além da identificação do seu lema, recorrendo às ferramentas descritas na secção 3.1.

O sistema Reciclagem também participou na tarefa de inferência textual. Neste caso, recorrendo exclusivamente aos *synsets* e relações de hiperonímia do CONTO.PT. Ao contrário dos valores de similaridade calculados, esta previsão de inferência textual não foi utilizada pela abordagem ASAPP. A classificação de inferência é relativamente simples e baseia-se em três parâmetros principais:

- δ, a proporção mínima de palavras que a frase t pode ter a mais ou menos que a frase h.
- θ_s, ponto de corte nas pertenças dos synsets, isto é, todas as palavras com pertença inferior a θ_s são removidas do respectivo synset.
- θ_h, ponto de corte na confiança das relações de hiperonímia, isto é, todas as relações de hiperonímia com confiança inferior a θ_h são ignoradas.

Inicialmente, é calculada a diferença absoluta entre o número de palavras de classe aberta nas frases t e h. Se esse valor for superior a δ, considera-se que não há inferência. Caso contrário, aplica os pontos de corte e usa-se a (sub-)*wordnet* resultante. Depois:

- Se todas as palavras de t estiverem em h, ou tiverem um sinónimo em h, as frases são consideradas paráfrases (*Paraphrase*);
- Se, por outro lado, todas as palavras de t tiverem um sinónimo ou um hiperónimo em h, considera-se que uma frase é inferência da outra (*Entailment*).
- Se nenhuma das condições anteriores se verificar, considera-se que não há qualquer tipo de inferência.

6 ASAPP

Na classificação, na regressão, no conjunto de classificadores, na selecção de características, entre outros, o sistema ASAPP utiliza a ferramenta Weka (Hall et al., 2009) para aprender, de forma

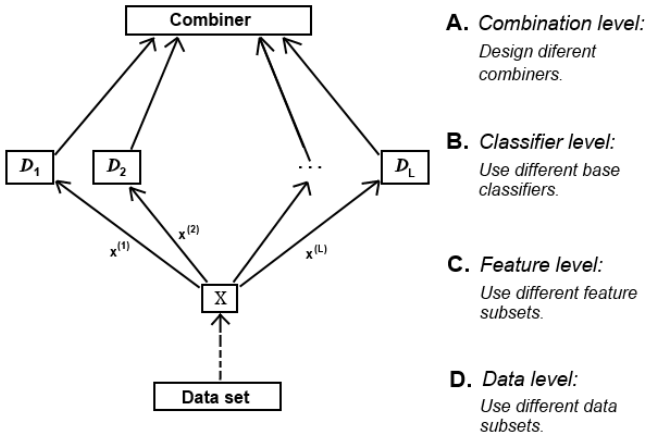

Figura 1: Abordagens para criar conjuntos de classificadores/modelos de regressão (em inglês *ensembles*) (Kuncheva, 2004)

supervisionada, a análise de regressão da similaridade e a classificação das três categorias de inferência textual (Paráfrase, Inferência Textual ou Nenhuma relação). Weka é uma grande coleção de algoritmos de aprendizagem implementados na linguagem de programação Java e continuamente em actualização. Por isso, inclui grande parte dos algoritmos mais recentes que representam o estado da arte da aprendizagem automática.

Seja a aprender, a classificar inferência textual, ou a calcular a similaridade entre frases, um conjunto de classificadores ou modelos de regressão geralmente tem melhor desempenho que um isolado (Kuncheva, 2004). Há quatro abordagens normalmente adotadas para criar conjuntos em aprendizagem (ver figura 1), cada uma focada num diferente nível de ação. A abordagem A considera as diferentes formas de combinar os resultados dos classificadores ou modelos de regressão, mas não existe uma evidência que esta estratégia seja melhor do que usar diferentes modelos (Abordagem B). Quanto às características (Abordagem C), diferentes subconjuntos podem ser usados para treinar classificadores (ou modelos regressão), sendo que estes possam utilizar o mesmo algoritmo de classificação (ou regressão) ou não. Finalmente, a coleção pode ser repartida de forma a que cada classificador (ou modelo de regressão) possa ser treinado no seu próprio conjunto de dados (Abordagem D).

Na criação do sistema ASAPP, foram seguidas as três primeiras abordagens de criação de conjuntos de classificadores e modelos de regressão, já que a nível dos dados (Abordagem D), o conjunto foi sempre o mesmo – aquele fornecido pela coleção ASSIN para o treino – , com validação cruzada através de 10 conjuntos (*10-fold cross-validation*). As características utilizadas foram todas as apresentadas no capítulo 4.

Utilizando a abordagem A, duas das configurações submetidas foram resultado da combinação da classificação de inferência textual obtida por diferentes classificadores (três classificadores num caso e cinco noutro) e foi escolhido o resultado final por *Maioria de Votos* (Kittler et al., 1998) para cada par de frases.

Pela abordagem B por duas vezes, ao combinarmos diferentes modelos, como os de regressão para a similaridade, utilizou-se em uma das configurações uma técnica conhecida por *Boosting* que iterativamente cria um modelo melhor com base no desempenho do modelo criado anteriormente (Friedman, 1999). Em outra configuração submetida para a similaridade, foi selecionado automaticamente o classificador com melhor desempenho, ou seja, que apresentava o menor erro quadrático médio (*mean-squared error*).

A abordagem C foi seguida na terceira configuração submetida para a inferência textual, onde um conjunto de características é selecionado automaticamente, desde que tenham pouca correlação entre si, mas uma alta correlação com a classe a prever, antes do treino efetivo.

Como última submissão para a similaridade, foi utilizado um processo gaussiano (Mackay, 1998) implementado no Weka de forma simplificada sem afinação por hiper-parâmetros.

Em resumo, a tabela 1 apresenta todos os algoritmos utilizados em cada configuração submetida e respetivamente para cada tarefa em foco: inferência e similaridade textual. É de notar que se procurou utilizar para cada configuração o mesmo conjunto de algoritmos para treinar os modelos em ambas variantes: Português-Europeu e Português-Brasileiro, tendo apenas sido utilizado em cada caso a coleção própria de cada variante da língua portuguesa.

7 Discussão de Resultados

De forma a comparar a abordagem baseada em conhecimento, Reciclagem, com a abordagem supervisionada, ASAPP, são de seguida apresentados os resultados obtidos por cada sistema no âmbito da sua participação na tarefa ASSIN. Os cálculos do coeficiente de correlação de Pearson para a similaridade, do erro quadrático médio (MSE) e da exatidão da inferência textual foram efetuados a partir do *script* disponibilizado pela organização da tarefa.

Configuração	Inferência	Similaridade
	Algoritmo específico do Weka utilizado para cada tarefa	
1	Voto por maioria de 3 classificadores (Kittler et al., 1998; Kuncheva, 2004) `weka.classifiers.meta.Vote -S 1 -R AVG -B (3 classificadores...)`	Regressão Aditiva por *Boosting* (Friedman, 1999) `weka.classifiers.meta.AdditiveRegression -S 1.0 -I 10 -W weka.classifiers.meta.RandomSubSpace --- -P 0.5 -S 1 -I 10 -W weka.classifiers.trees.REPTree --- -M 2 -V 0.0010 -N 3 -S 1 -L -1`
2	Voto por maioria de 5 classificadores (Kittler et al., 1998; Kuncheva, 2004) `weka.classifiers.meta.Vote -S 1 -R AVG -B (5 classificadores...)`	Esquema Múltiplo de Seleção (Hall et al., 2009) `weka.classifiers.meta.MultiScheme -X 0 -S 1 -B` (5 modelos de regressão...)
3	Redução Automática de Características (Hall et al., 2009) `weka.classifiers.meta.AttributeSelectedClassifier -E ‘‘weka.attributeSelection.CfsSubsetEval¨-S ‘‘weka.attributeSelection.BestFirst -D 1 -N 5" -W weka.classifiers.trees.J48 --- -C 0.25 -M 2`	Processo Gaussiano Simples (Mackay, 1998) `weka.classifiers.functions.GaussianProcesses -L 1.0 -N 0 -K ‘‘weka.classifiers.functions.supportVector.NormalizedPolyKernel -C 250007 -E 2.0"`

Tabela 1: Configurações submetidas (submissões) e como foram treinadas.

7.1 Resultados de similaridade para diferentes configurações Reciclagem

No sistema Reciclagem, podemos dizer que uma configuração para calcular a similaridade entre duas frases tem pelo menos dois parâmetros – rede semântica e heurística. No caso de se utilizar uma heurística baseada na semelhança de vizinhanças, pode ainda variar o método de obter as vizinhanças ($Total$, $m \times n$ e $Max(m \times n)$). No entanto, verificamos empiricamente que os resultados obtidos com vizinhanças calculadas pelo método $Max(m \times n)$ batiam consistentemente os restantes. Já ao se utilizar a wordnet difusa CONTO.PT, podem variar-se parâmetros ao nível da consideração da pertença de cada palavra, do ponto de corte a aplicar sobre a pertença das palavras aos *synsets*, ou sobre a confiança das relações de hiperonímia, e ainda o peso a dar a cada relação (ρ).

Para além da utilização individual de cada uma das redes apresentadas na secção 3.2, foi criada uma rede com os triplos de todos os recursos e outra baseada na redundância, com os triplos que ocorriam em pelo menos três recursos (*Redun3*). No entanto, a primeira acabou por não ser utilizada porque, devido a ser muito grande, tornava os cálculos mais demorados, para além de se ter verificado empiricamente que não levava a melhores resultados que, por exemplo, a rede baseada em redundância.

Numa avaliação que recorreu às coleções de treino, a forma de calcular a similaridade que levou a um coeficiente de Pearson mais elevado foi, sem qualquer exceção, a $Max(M \times n)$. Este comportamento foi posteriormente confirmado na coleção de teste. Assim, todos os resultados mostrados nesta seção foram calculados dessa forma. No caso da CONTO.PT, foram utilizados os seguintes parâmetros:

- Pertença mínima da palavra a um *synset*: $min(\mu(p, synsets)) = 0.05$
- Corte aplicado nos *synsets*: $corte_{synsets} = 0.05$
- Peso multiplicado pela pertença num *synset*: $rho_s = 1$
- Peso multiplicado pela confiança numa relação de hiperonímia: $rho_h = 0.1$
- Peso multiplicado pela confiança numa outra relação: $rho_o = 0.02$

As tabelas 2 e 3 mostram as configurações que obtiveram melhor classificação na coleção de

Rede	Sim Frase	Métrica	Pearson	MSE
Redun3	$Max(m \times n)$	Overlap	0,600	1,173
Redun3	$Max(m \times n)$	Dice	0,598	1,185
OpenWN-PT	$Max(m \times n)$	Jaccard	0,596	1,159
Redun3	$Max(m \times n)$	Jaccard	0,596	1,190
PAPEL	$Max(m \times n)$	Overlap	0,594	1,195
TeP	$Max(m \times n)$	Dice	0,592	1,330
PULO	$Max(m \times n)$	Jaccard	0,590	1,259
OpenWN-PT	N/A	PPR	0,528	1,301
CONTO.PT	N/A		0,587	1,189

Tabela 2: Melhores configurações e configurações selecionadas de rede semântica + métrica para similaridade na coleção de treino PT-PT.

Rede	Sim Frase	Métrica	Pearson	MSE
Redun3	$Max(m \times n)$	Overlap	0,546	1,065
OpenWN-PT	$Max(m \times n)$	Dice	0,546	1,077
OpenWN-PT	$Max(m \times n)$	Jaccard	0,545	1,081
OpenWN-PT	$Max(m \times n)$	Overlap	0,544	1,039
Redun3	$Max(m \times n)$	Jaccard	0,544	1,070
Redun3	$Max(m \times n)$	Overlap	0,544	1,052
PAPEL	$Max(m \times n)$	Overlap	0,543	1,027
TeP	$Max(m \times n)$	Dice	0,543	1,090
PULO	$Max(m \times n)$	Jaccard	0,541	1,037
PAPEL	N/A	PPR	0,447	1,150
CONTO.PT	N/A		0,535	1,078

Tabela 3: Melhores configurações e configurações selecionadas de rede semântica + métrica para similaridade na coleção de treino PT-BR.

treino, identificando a rede, a heurística, o valor do coeficiente de Pearson e ainda do erro quadrático médio (MSE). Cada tabela inclui ainda uma pequena selecção com os melhores resultados que usam redes ou heurísticas não contemplados nos anteriores. As tabelas 4 e 5 apresentam os mesmos resultados, mas para a coleção de teste.

A observação dos resultados mostra que a diferença entre as melhores configurações para cada rede é ténue, sendo muitas vezes necessário recorrer à terceira casa decimal do coeficiente de Pearson. Isto mostra que a heurística aplicada acaba por ser mais relevante que o conteúdo da própria rede. Por exemplo, os melhores resultados foram sempre obtidos pelo coeficiente Dice, a distância média levou sempre a resultados muito baixos, aqui não apresentados, enquanto que o Personalized PageRank ficou sempre abaixo alguns pontos que as heurísticas baseadas na semelhança de conjuntos. Ainda assim, as últimas heurísticas mereciam uma melhor afinação que acabou por não ser realizada.

Rede	Sim Frase	Métrica	Pearson	MSE
Redun3	$Max(m \times n)$	Overlap	0,536	1,105
Redun3	$Max(m \times n)$	Dice	0,536	1,130
Redun3	$Max(m \times n)$	Jaccard	0,535	1,149
OpenWN-PT	$Max(m \times n)$	Jaccard	0,533	1,141
TeP	$Max(m \times n)$	Dice	0,532	1,131
TeP	$Max(m \times n)$	Jaccard	0,532	1,151
PAPEL	$Max(m \times n)$	Dice	0,530	1,146
PULO	$Max(m \times n)$	Jaccard	0,527	1,313
OpenWN-PT	N/A	PPR	0,513	1,177
CONTO.PT	N/A		0,526	1,179

Tabela 4: Melhores configurações e configurações selecionadas de rede semântica + métrica para similaridade na coleção de teste PT-PT.

Rede	Sim Frase	Métrica	Pearson	MSE
TeP	$Max(m \times n)$	Overlap	0,593	1,256
OpenWN-PT	$Max(m \times n)$	Dice	0,589	1,312
OpenWN-PT	$Max(m \times n)$	Overlap	0,589	1,345
TeP	$Max(m \times n)$	Dice	0,588	1,311
OpenWN-PT	$Max(m \times n)$	Jaccard	0,588	1,329
Redun3	$Max(m \times n)$	Dice	0,588	1,356
PULO	$Max(m \times n)$	Dice	0,584	1,326
PAPEL	$Max(m \times n)$	Dice	0,584	1,335
OpenWN-PT	N/A	PPR	0,464	1,225
CONTO.PT	N/A		0,580	1,367

Tabela 5: Melhores configurações e configurações selecionadas de rede semântica + métrica para similaridade na coleção de teste PT-BR.

Apesar desta abordagem não depender de um treino prévio, verifica-se uma curiosidade: enquanto que, nas coleções de treino, os resultados obtidos para o coeficiente de Pearson eram, de uma forma geral, superiores para o PT-PT (cerca de 0,6 contra 0,54), nas coleções de teste esta tendência inverteu-se (cerca de 0,53 contra 0,59).

Apesar de tudo, é possível especular um pouco sobre o desempenho das redes. Por exemplo, confirma-se que a combinação das sete redes (Redun3) leva consistentemente a bons resultados, e só não obtém os melhores resultados na coleção de teste para PT-BR. Relativamente a redes individuais, a OpenWN-PT destaca-se por aparecer sempre entre as melhores. E apesar de ter sido criada para o português do Brasil e de se limitar a cobrir relações de sinonímia e antonímia, a rede TeP teve um desempenho superior nas coleções de teste, inclusivamente com o melhor resultado para o PT-BR. Por fim, apesar de nunca se chegar aos melhores resultados, a utilização do CONTO.PT leva a resultados que ficam apenas entre uma e duas décimas abaixo dos melhores. Sendo uma rede criada recentemente, pouco explorada, e onde foi aplicada uma heurística que também deveria ter sido alvo de uma afinação mais profunda, vemos os seus resultados como promissores.

	θ_s	θ_h	δ	**Exatidão**	**Macro F1**
PT-PT	0,1	0,01	0,5	73,83%	0,45
	0,1	0,1	0,4	71,67%	0,38
	0,25	0,2	0,5	73,83%	0,45
PT-BR	0,1	0,01	0,3	77,47%	0,31
	0,1	0,1	0,5	76,70%	0,42
	0,2	0,2	0,1	77,70%	0,29

Tabela 6: Resultados da inferência textual na coleção de treino com a abordagem Reciclagem.

	θ_s	θ_h	δ	**Exatidão**	**Macro F1**
PT-PT	0,05	0,01	0,3	70,80%	0,32
	0,1	0,1	0,5	73,10%	0,43
	0,15	0,1	0,4	72,10%	0,38
PT-BR	0,1	0,01	0,3	78,30%	0,33
	0,15	0,1	0,3	79,05%	0,39
	0,2	0,2	0,1	77,65%	0,29

Tabela 7: Resultados da inferência textual na coleção de teste com a abordagem Reciclagem.

7.2 Resultados para a inferência textual Reciclagem

As tabelas 6 e 7 apresentam os resultados de algumas configurações da abordagem Reciclagem para a inferência textual, respetivamente nas coleções de treino e teste. Para além dos valores da exatidão e Macro F1, são apresentados os valores dos parâmetros utilizados, nomeadamente os pontos de corte θ_s e θ_h, e ainda a proporção δ.

Olhando apenas para a exatidão, os valores nesta tarefa são bastante aceitáveis e, como se verá na próxima seção, mais próximos da abordagem supervisionada. Por outro lado, o valor da Macro F1 é inferior a 0,5, e por isso menos promissor. Tanto no treino como teste, a exatidão é superior para o PT-BR. No entanto, constatou-se que a coleção PT-PT tinha mais casos de inferência que a PT-BR, o que dificulta a tarefa para esta variante. Mais propriamente, cerca de 24% dos pares na coleção de treino PT-PT eram casos de inferência e cerca de 7% de paráfrase, proporções que descem para cerca de 17% e 5% em PT-BR. Ou seja, um sistema que, no caso do PT-BR, respondesse sempre que não existia inferência, iria obter cerca de 78% de exatidão, ainda que com impacto negativo na Macro F1. Olhando apenas para a Macro F1, os resultados para PT-PT são ligeiramente superiores a PT-BR.

7.3 Resultados para diferentes configurações ASAPP

A avaliação que recorreu às coleções de treino para criar modelos de classificadores e de regressão para as respetivas tarefas de inferência e similaridade é apresentada na tabela 8.

Submissão	**Inferência exatidão**	**F1**	**Similaridade Pearson**	**MSE**
1 - PTBR	79,87%	**0,767**	0,620	0,677
1 - PTPT	78,27%	**0,766**	0,715	0,613
2 - PTBR	**80,77%**	0,765	0,622	0,677
2 - PTPT	**78,73%**	0,765	0,716	0,612
3 - PTBR	76,50%	0,759	**0,635**	**0,668**
3 - PTPT	77,77%	0,775	**0,723**	**0,606**

Tabela 8: Melhores configurações e configurações selecionadas para submissão com base no resultado de validação cruzada do treino.

Após a divulgação dos resultados de teste pela organização do ASSIN (tabela 9), foi comprovado que tanto na fase de treino como na de teste, a submissão 2 (Maioria de votos entre 5 classificadores) apresenta melhores resultados de exatidão para a classificação da inferência textual, conseguindo-se uma exatidão de 80,77% para o Português Brasileiro com um MSE de 0,765, e de 78,73% e MSE 0,765 para o Português Europeu.

Esta coerência também é verificada na similaridade, uma vez que a terceira submissão (Processo Gaussiano) apresenta resultados idênticos à primeira na fase de testes, mas ultrapassa-a em muito na fase de treino. Este algoritmo é atualmente oferecido por outras frameworks de uma forma muito mais completa e com possibilidade de estudo da redução de características de forma integrada, como é o caso do Simulink em Matlab[12]. Como possível melhoria, pretende-se explorar variantes deste algoritmo com a adoção desta ferramenta.

Quanto às características importa realçar que algumas acabaram por não ser devidamente exploradas, nomeadamente a comparação de n-gramas, e as características distribucionais obtidas a partir de modelação de tópicos (*topic modeling*), propostas inicialmente pelas anteriores versões do ASAP, para o Inglês (Alves et al., 2014, 2015).

De modo a evitar um aumento do tempo que o treino irá demorar com este acrescento de novas características e de forma a perceber a contribuição de cada uma em particular no processo de aprendizagem, um possível melhoramento será um estudo de selecção de características com base na sua relevância.

[12] http://www.mathworks.com/products/simulink/?requestedDomain=www.mathworks.com

Submissão	Inferência exatidão	F1	Similaridade Pearson	MSE
1 - PTBR	81,20%	0.5	**0,65**	**0,44**
1 - PTPT	77,75%	0.57	**0,68**	**0,70**
2 - PTBR	**81,56%**	0,47	0,65	0,44
2 - PTPT	**78,90%**	0,58	0,67	0,71
3 - PTBR	77,10%	**0,5**	0,65	0,44
3 - PTPT	74,35%	**0,59**	0,68	0,73

Tabela 9: Resultado final do teste das tarefas de inferência e similaridade pela organização do ASSIN.

8 Conclusões

Foram apresentadas duas abordagens distintas à tarefa de avaliação conjunta ASSIN: uma primeira, apelidada de Reciclagem, baseada exclusivamente em heurísticas sobre redes semânticas para a língua portuguesa; e uma segunda, apelidada de ASAPP, baseada em aprendizagem automática supervisionada.

De forma a aproveitar um conjunto de recursos e ferramentas existentes para o processamento computacional do português, foram apresentadas redes semânticas e ferramentas que estão acessíveis à comunidade. A partir destes recursos extraíram-se características distintas para implementar as duas abordagens que participaram na tarefa ASSIN.

Após comparação com os resultados da coleção dourada, verificou-se que a abordagem ASAPP supera a abordagem Reciclagem de forma consistente. Isto ocorre tanto para o Português Europeu como para o Português Brasileiro, onde o desempenho atinge uma exatidão de $80,28\% \pm 0.019$ para a inferência textual, enquanto que a correlação dos valores atribuídos para a similaridade semântica com aqueles atribuídos por humanos é de $66,5\% \pm 0.021$.

Por outro lado, através da abordagem Reciclagem verificou-se que é possível obter valores semelhantes através da exploração de diferentes redes, apesar daquela que mais se destacou resultar da combinação das sete redes usadas.

9 Trabalho Futuro

O trabalho aqui apresentado refere-se a uma abordagem inicial à tarefa ASSIN, sujeita às restrições temporais da avaliação, onde agora nos apercebemos que quisemos experimentar e comparar demasiadas abordagens. Após o período da avaliação, identificamos vários aspetos a melhorar na extração de algumas características, para além de novas características a extrair em abordagens futuras.

Por exemplo, entre as experiências entretanto realizadas na abordagem Reciclagem, sobre a coleção de treino, verificámos que o cálculo da similaridade em dois passos – primeiro, intersecção de lemas, depois, aplicação da heurística $Max(m \times n)$ sobre os lemas não partilhados pelas duas frases – leva a melhorias significativas de desempenho, tanto temporal como qualitativo. Na verdade, uma heurística baseada exclusivamente na interseção de lemas seria suficiente para ultrapassar os resultados obtidos pelo sistema Reciclagem em cerca de 0,1 pontos no coeficiente de Pearson. A aplicar, estas melhorias terão também como consequência a melhoria dos resultados da abordagem ASAPP.

Entre características que pretendemos explorar no futuro, destacamos as características distribucionais, quer as obtidas a partir de modelação de tópicos (*topic modeling*), propostas inicialmente pelas anteriores versões do ASAP, para o Inglês (Alves et al., 2014, 2015), quer as baseadas em *word embeddings* (Mikolov et al., 2013).

Contudo, uma descrição mais aprofundada das novas abordagens a esta tarefa está fora do âmbito deste artigo e será o alvo de uma publicação futura.

Agradecimentos

Este trabalho é parcialmente financiado por Fundos FEDER através do Programa Operacional Factores de Competitividade — COMPETE e por Fundos Nacionais através da FCT — Fundação para a Ciência e a Tecnologia no âmbito do projeto Relevance Mining and Detection System (REMINDS) Ref. UTAP-ICDT/EEI-CTP/0022/2014

Referências

Afonso, Susana, Eckhard Bick, Renato Haber & Diana Santos. 2001. Floresta Sintá(c)tica: um "Treebank" para o Português. Em Anabela Gonçalves & Clara Nunes Correia (eds.), *Actas do XVII Encontro Nacional da Associação Portuguesa de Linguística*, 533–545.

Agirre, Eneko, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria & Janyce Wiebe. 2015. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. Em *Proceedings of the 9th internatio-*

nal workshop on semantic evaluation (SemEval 2015), 252–263.

Agirre, Eneko, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau & Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. Em *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, 81–91.

Agirre, Eneko, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre & Weiwei Guo. 2013. *sem 2013 shared task: Semantic textual similarity. Em *Proceedings of 2nd Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, 32–43. ACL Press.

Agirre, Eneko, Mona Diab, Daniel Cer & Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. Em *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, 385–393. ACL Press.

Agirre, Eneko, Oier Lopez De Lacalle & Aitor Soroa. 2009. Knowledge-based WSD on specific domains: performing better than generic supervised WSD. Em *Proceedings of 21st International Joint Conference on Artificial Intelligence* IJCAI 2009, 1501–1506. Morgan Kaufmann Publishers Inc.

Agirre, Eneko & Aitor Soroa. 2009. Personalizing PageRank for word sense disambiguation. Em *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* EACL'09, 33–41. ACL Press.

Alves, Ana, David Simões, Hugo Gonçalo Oliveira & Adriana Ferrugento. 2015. Asap-ii: From the alignment of phrases to textual similarity. Em *Proceedings of 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 184–189. ACL Press.

Alves, Ana O., Adriana Ferrugento, Mariana Lourenço & Filipe Rodrigues. 2014. Asap: Automatic semantic alignment for phrases. Em *SemEval Workshop, COLING 2014, Ireland*, 104–108.

Androutsopoulos, Ion & Prodromos Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. *J. Artif. Int. Res.* 38(1). 135–187.

Banerjee, Satanjeev & Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. Em *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, 805–810.

Barreiro, Anabela. 2008. Paramt: A paraphraser for machine translation. Em *Computational Processing of the Portuguese Language: 8th International Conference*, 202–211.

Fellbaum, Christiane (ed.). 1998. *WordNet: An Electronic Lexical Database (language, speech, and communication)*. The MIT Press.

Fonseca, Evandro B., Gabriel C. Chiele & Aline A. Vanin. 2015. Reconhecimento de Entidades Nomeadas para o Portugues Usando o OpenNLP. Em *Anais do Encontro Nacional de Inteligência Artificial e Computacional (ENIAC 2015)*, s. pp.

Freitas, Cláudia & Diana Santos. 2015. *Pesquisas e Perspectivas em Linguística de Corpus* chap. Blogs, Amazônia e a Floresta Sintá(c)tica: um Corpus de um novo Gênero?, 123–150. Mercado de Letras.

Friedman, J.H. 1999. Stochastic gradient boosting. Relatório técnico. Stanford University.

Gonçalo Oliveira, Hugo. 2016. CONTO.PT: Groundwork for the Automatic Creation of a Fuzzy Portuguese Wordnet. Em *Proceedings of 12th International Conference on Computational Processing of the Portuguese Language (PROPOR 2016)*, vol. 9727 LNAI, 283–295.

Gonçalo Oliveira, Hugo, Leticia Antón Pérez, Hernani Costa & Paulo Gomes. 2011. Uma rede léxico-semântica de grandes dimensões para o português, extraída a partir de dicionários electrónicos. *Linguamática* 3(2). 23–38.

Gonçalo Oliveira, Hugo, Inês Coelho & Paulo Gomes. 2014. Exploiting Portuguese lexical knowledge bases for answering open domain cloze questions automatically. Em *Proceedings of the 9th Language Resources and Evaluation Conference* LREC 2014, ELRA.

Gonçalo Oliveira, Hugo, Valeria de Paiva, Cláudia Freitas, Alexandre Rademaker, Livy Real & Alberto Simões. 2015. As wordnets do português. Em Alberto Simões, Anabela Barreiro, Diana Santos, Rui Sousa-Silva & Stella E. O. Tagnin (eds.), *Linguística, Informática e Tradução: Mundos que se Cruzam*, vol. 7(1)

OSLa: Oslo Studies in Language, 397–424. University of Oslo.

Gonçalo Oliveira, Hugo, Diana Santos, Paulo Gomes & Nuno Seco. 2008. PAPEL: A dictionary-based lexical ontology for Portuguese. Em *Proceedings of Computational Processing of the Portuguese Language – 8th International Conference (PROPOR 2008)*, vol. 5190 LNCS/LNAI, 31–40.

Hall, Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann & Ian H. Witten. 2009. The weka data mining software: An update. *SIGKDD Explor. Newsl.* 11(1). 10–18.

Jiang, Jay J. & David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. Em *Proc. of the Int'l. Conf. on Research in Computational Linguistics*, 19–33.

Kittler, J., M. Hatef, Robert P.W. Duin & J. Matas. 1998. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(3). 226–239.

Kuncheva, Ludmila I. 2004. *Combining pattern classifiers: Methods and algorithms*. Wiley-Interscience.

Lesk, Michael. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. Em *Proceedings of the 5th Annual International Conference on Systems Documentation (SIGDOC '86)*, 24–26.

Mackay, David J.C. 1998. *Introduction to gaussian processes*. Dept. of Physics, Cambridge University, UK.

Maziero, Erick G., Thiago A. S. Pardo, Ariani Di Felippo & Bento C. Dias-da-Silva. 2008. A Base de Dados Lexical e a Interface Web do TeP 2.0 - Thesaurus Eletrônico para o Português do Brasil. Em *VI Workshop em Tecnologia da Informação e da Linguagem Humana (TIL)*, 390–392.

Mikolov, Tomas, Kai Chen, Greg Corrado & Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv CoRR* arXiv:1301.3781.

Mota, Cristina. 2007. *Reconhecimento de entidades mencionadas em português: Documentação e actas do HAREM, a primeira avaliação conjunta na área* chap. Estudo Preliminar para a avaliação de REM em Português, 19–34. Linguateca.

de Paiva, Valeria, Alexandre Rademaker & Gerard de Melo. 2012. OpenWordNet-PT: An open Brazilian wordnet for reasoning. Em *Proceedings of 24th International Conference on Computational Linguistics* COLING (Demo Paper), 353–360.

Pinheiro, Vladia, Vasco Furtado & Adriano Albuquerque. 2014. Semantic textual similarity of portuguese-language texts: An approach based on the semantic inferentialism model. Em *Computational Processing of the Portuguese Language - 11th International Conference, PROPOR 2014, São Carlos/SP, Brazil, October 6-8, 2014. Proceedings*, 183–188.

Ponzetto, Simone Paolo & Roberto Navigli. 2010. Knowledge-rich word sense disambiguation rivaling supervised systems. Em *Proceedings of 48th Annual Meeting of the Association for Computational Linguistics* ACL 2012, 1522–1531. ACL Press.

Rodrigues, Ricardo, Hugo Gonçalo-Oliveira & Paulo Gomes. 2014. LemPORT: a High-Accuracy Cross-Platform Lemmatizer for Portuguese. Em Maria João Varanda Pereira, José Paulo Leal & Alberto Simões (eds.), *Proceedings of the 3rd Symposium on Languages, Applications and Technologies (SLATE '14)* OpenAccess Series in Informatics, 267–274.

Rychalska, Barbara, Katarzyna Pakulska, Krystyna Chodorowska, Wojciech Walczak & Piotr Andruszkiewicz. 2016. Samsung poland NLP team at semeval-2016 task 1: Necessity for diversity; combining recursive autoencoders, wordnet and ensemble methods to measure semantic similarity. Em *Proceedings of the 10th International Workshop on Semantic Evaluation*, 602–608.

Seno, Eloize Rossi Marques & Maria das Graças Volpe Nunes. 2008. Some experiments on clustering similar sentences of texts in portuguese. Em *Computational Processing of the Portuguese Language, 8th International Conference*, 133–142.

Simões, Alberto & Xavier Gómez Guinovart. 2014. Bootstrapping a Portuguese wordnet from Galician, Spanish and English wordnets. Em *Advances in Speech and Language Technologies for Iberian Languages*, vol. 8854 LNCS, 239–248.

Simões, Alberto, Álvaro Iriarte Sanromán & José João Almeida. 2012. Dicionário-Aberto: A source of resources for the Portuguese language processing. Em *Proceedings of 10th International Conference on the Computational*

Processing of the Portuguese Language (PROPOR 2012), vol. 7243 LNCS, 121–127.

Sultan, Md Arafat, Steven Bethard & Tamara Sumner. 2015. Dls@cu: Sentence similarity from word alignment and semantic vector composition. Em *Proc. of SemEval 2015*, 148–153. ACL.