Universidade do Minho

Universidade de Vigo

linguaMATICA

Volume 8, Número 2- Dezembro 2016

ISSN: 1647-0818

Volume 8, Número 2 – Dezembro 2016

LinguaMÁTICA

ISSN: 1647–0818

Conteúdo

Editorial

Neste oitavo ano de vida a Linguamática teve, como é seu hábito, duas edições. A primeira, em Julho, apenas com três artigos, e esta edição, de Dezembro, com artigos alargados correspondentes a uma workshop realizada com conjunto sim a conferência PROPOR'2016, a ASSIN: Avaliação de Similaridade Semântica e de Inferência Textual. Se por um lado o número de artigos publicado é pequeno, por outro, o número de artigos recebidos para avaliação não o foi, estando dentro da média habitual da Linguamática.

Mas o ano de 2016, embora mau, como sabemos, para muitas personalidades do mundo da música, não o foi para a Linguamática, que continua a ser indexada pela Scopus (embora o site da Scopus ainda não inclua todas as edições recentes), e passou a ser incluída na Web of Science da Thomson Reuters, no índice ESCI (Emerging Sources Citation Index), um índice de revistas selecionadas para avaliação e possível integração nos índices de topo. São, sem dúvidas, duas notícias que nos fazem muito orgulhosos dos nossos autores.

Xavier Gómez Guinovart
José João Almeida
Alberto Simões

Prólogo

Avaliação de Similaridade Semântica e de Inferência Textual

A Avaliação de Similaridade Semântica e de Inferência Textual (ASSIN) foi proposta como um Workshop em paralelo com o PROPOR 2016 para apresentação dos resultados da avaliação conjunta de duas subtarefas relacionadas, tratando da língua portuguesa, especificamente do Português do Brasil (PB) e Europeu (PE). Ambas as subtarefas dizem respeito ao entendimento de um par de sentenças: a similaridade semântica (STS, Semantic Textual Similarity) é uma medida numérica de 1 a 5 do quão similar é o conteúdo das duas sentenças; e a inferência textual (RTE, Recognizing Textual Entailment) consiste em classificar o par como tendo uma relação de implicação, paráfrase, ou nenhuma das duas.

A avaliação conjunta deixou como legado o córpus ASSIN de 10.000 pares de sentenças (5.000 em PB e 5.000 em PE) usado pelos participantes e que está publicamente disponível em `http://nilc.icmc.usp.br/assin/`. *Somos gratos a todos os anotadores do córpus, pois sem eles a avaliação não teria sido realizada.*

Foram seis os participantes da avaliação: três do Brasil (Solo Queue, Blue Man Group, LEC-UNIFOR) e três de Portugal (INESC-ID, ASAPP, Reciclagem) sendo que todos participaram da tarefa STS, e somente quatro deles da tarefa RTE.

Nesta edição especial da Linguamática em homenagem ao Workshop ASSIN, trazemos o artigo com a apresentação da Avaliação Conjunta e mais cinco versões revisadas e estendidas dos seguintes artigos apresentados no Workshop, sendo que as equipes ASAPP e Reciclagem escreveram um único artigo reportando ambos os resultados.

Desejamos a todos uma leitura proveitosa destes trabalhos!

Erick Fonseca
Leandro Santos
Marcelo Criscuolo
Sandra Aluísio

Comissão Científica

Alberto Álvarez Lugrís,
Universidade de Vigo

Alberto Simões,
Universidade do Minho

Aline Villavicencio,
Universidade Federal do Rio Grande do Sul

Álvaro Iriarte Sanroman,
Universidade do Minho

Ana Frankenberg-Garcia,
University of Surrey

Anselmo Peñas,
Univers. Nac. de Educación a Distancia

Antón Santamarina,
Universidade de Santiago de Compostela

Antoni Oliver González,
Universitat Oberta de Catalunya,

Antonio Moreno Sandoval,
Universidad Autónoma de Madrid

António Teixeira,
Universidade de Aveiro

Arantza Díaz de Ilarraza,
Euskal Herriko Unibertsitatea

Arkaitz Zubiaga,
Dublin Institute of Technology

Belinda Maia,
Universidade do Porto

Carmen García Mateo,
Universidade de Vigo

Diana Santos,
Linguateca/Universidade de Oslo

Ferran Pla,
Universitat Politècnica de València

Gael Harry Dias,
Université de Caen Basse-Normandie

Gerardo Sierra,
Univers. Nacional Autónoma de México

German Rigau,
Euskal Herriko Unibertsitatea

Helena de Medeiros Caseli,
Universidade Federal de São Carlos

Horacio Saggion,
University of Sheffield

Hugo Gonçalo Oliveira,
Universidade de Coimbra

Iñaki Alegria,
Euskal Herriko Unibertsitatea

Irene Castellón Masalles,
Universitat de Barcelona

Joaquim Llisterri,
Universitat Autònoma de Barcelona

José João Almeida,
Universidade do Minho

José Paulo Leal,
Universidade do Porto

Joseba Abaitua,
Universidad de Deusto

Juan-Manuel Torres-Moreno,
Lab. Informatique d'Avignon - UAPV

Kepa Sarasola,
Euskal Herriko Unibertsitatea

Laura Plaza,
Complutense University of Madrid

Lluís Padró,
Universitat Politècnica de Catalunya

Marcos Garcia,
Universidade de Santiago de Compostela

María Inés Torres,
Euskal Herriko Unibertsitatea

Maria das Graças Volpe Nunes,
Universidade de São Paulo

Mercè Lorente Casafont,
Universitat Pompeu Fabra

Mikel Forcada,
Universitat d'Alacant

Pablo Gamallo Otero,
Universidade de Santiago de Compostela

Patrícia Cunha França,
Universidade do Minho

Rui Pedro Marques,
Universidade de Lisboa

Salvador Climent Roca,
Universitat Oberta de Catalunya

Susana Afonso Cavadas,
University of Sheffield

Tony Berber Sardinha,
Pontifícia Univ. Católica de São Paulo

Xavier Gómez Guinovart,
Universidade de Vigo

Avaliação de Similaridade Semântica e de Inferência Textual

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

Visão Geral da Avaliação de Similaridade Semântica e Inferência Textual

Overview of the Evaluation of Semantic Similarity and Textual Inference

Erick Rocha Fonseca
Universidade de São Paulo
erickrf@icmc.usp.br

Leandro Borges dos Santos
Universidade de São Paulo
leandrobs@usp.br

Marcelo Criscuolo
Universidade de São Paulo
mcrisc@icmc.usp.br

Sandra Maria Aluísio
Universidade de São Paulo
sandra@icmc.usp.br

Resumo

Inferência Textual e Similaridade Semântica são duas tarefas do processamento de línguas naturais que tratam de pares de trechos de textos. O objetivo da primeira é determinar se o significado de um trecho implica o outro, enquanto que a segunda atribui uma pontuação de similaridade semântica ao par. Esse artigo apresenta os resultados da avaliação conjunta ASSIN (Avaliação de Similaridade Semântica e Inferência) e seu corpus, que foi anotado para ambas as tarefas nas variantes brasileira e europeia da língua portuguesa. O corpus difere de similares na literatura em suas três classes para a tarefa de inferência textual (Implicação, Paráfrase e Neutro) e por ter sido composto de sentenças extraídas de textos jornalísticos. Seis equipes participaram da avaliação conjunta, explorando diferentes estratégias.

Palavras chave

Avaliação conjunta, inferência textual, similaridade semântica

Abstract

Recognizing Textual Entailment and Semantic Textual Similarity are two natural language processing tasks dealing with pairs of text passages. The former aims to determine whether the meaning of one passage entails the other, while the latter assigns a semantic similarity score to the pair. This paper presents the results of the ASSIN shared task and its corpus, annotated for both tasks in the Brazilian and European varieties of the language. The corpus differs from similar ones in the literature in its three RTE classes (Entailment, Paraphrase and Neutral), and for having been composed of sentences extracted from newswire texts. Six teams took part in the shared task, exploring different strategies.

Keywords

Shared task, text entailment, semantic similarity

1 Introdução

A Avaliação de Similaridade Semântica e de Inferência Textual (ASSIN) foi proposta em paralelo com o PROPOR 2016, consistindo em duas subtarefas relacionadas. Ambas as subtarefas dizem respeito ao entendimento de um par de sentenças: a similaridade semântica (STS, *Semantic Textual Similarity*) (Agirre et al., 2015) é uma medida numérica de 1 a 5 do quão similar é o conteúdo das duas sentenças; e a inferência textual (RTE, *Recognizing Textual Entailment*) (Dagan et al., 2013) consiste em classificar o par como tendo uma relação de implicação, paráfrase, ou nenhuma das duas.

A definição exata destas tarefas não é universal. Outros conjuntos de dados apresentam escalas diferentes para a similaridade semântica (Agirre et al., 2015) ou a possibilidade de identificar contradição entre duas sentenças (Bentivogli et al., 2009). No caso do ASSIN, decidimos por uma escala de similaridade de 1 a 5 por achar mais fácil discriminar os diferentes níveis, enquanto na tarefa de inferência, nosso processo de criação de corpus não resultou em quase nenhum caso de contradição.

A avaliação ASSIN 2016 trouxe o primeiro corpus anotado para as duas tarefas em português, incluindo as variantes brasileira e europeia. Foram compiladas sentenças de textos reais, do gênero informativo (textos jornalísticos) em contraste com a abordagem utilizada para a construção de corpora similares em inglês, como SICK (Marelli et al., 2014) e SNLI (Bowman et al., 2015) e dos RTE Challenges (Bentivogli et al., 2009).

Aproveitamos os agrupamentos de notícias por assunto fornecidos pelo *Google News*[1] para

[1] https://news.google.com/

criar o corpus ASSIN 2016. Usamos modelos de espaço vetorial (Turney & Pantel, 2010) para selecionar sentenças similares de documentos diferentes, que passaram por um processo de filtragem manual (onde foram excluídos pares considerados ruidosos) e, por fim, foram anotados por juízes humanos. Cada par foi anotado por quatro pessoas com respeito às duas tarefas.

Participaram do ASSIN seis equipes, sendo três brasileiras e três portuguesas. Cada equipe participante pôde enviar até três saídas dos seus sistemas para cada combinação de variante e subtarefa. As seis equipes participaram da tarefa de similaridade semântica, e quatro delas participaram da inferência textual. É interessante notar que foram exploradas diferentes abordagens para tratar os problemas, mas nem todas foram capazes de superar os *baselines*.

Tratamos brevemente de avaliações conjuntas sobre as mesmas tarefas, para inglês, na Seção 2. Na Seção 3, apresentamos a definição detalhada das tarefas para o escopo do ASSIN 2016. Na Seção 4 descrevemos o processo de criação do corpus, assim como métricas usadas para a avaliação da concordância entre anotadores. Fornecemos também diretrizes para reduzir a subjetividade da anotação. A Seção 5 apresenta as seis equipes participantes e um resumo das suas abordagens. A Seção 6 descreve os *baselines* usados na tarefa e os resultados gerais. As conclusões e possíveis trabalhos futuros são apresentados na Seção 7.

2 Trabalhos Relacionados

A primeira competição de RTE foi o *PASCAL Recognising Textual Entailment Challenge* (RTE-1) (Dagan et al., 2005), que apresentou pares de sentenças coletados manualmente, tentando simular o cenário de aplicações de PLN. Por exemplo, em um cenário de Extração de Informação, a segunda sentença mencionava alguma propriedade de uma entidade mencionada na primeira. Nos anos seguintes, outras edições do evento foram realizadas, trazendo novos corpora anotados. Em particular, no RTE-4 (Giampiccolo et al., 2008), a avaliação trouxe a classificação de alguns pares como contradição. No SemEval 2014, foi utilizado o corpus SICK (Marelli et al., 2014), que trazia anotação tanto de RTE como de STS. Esta foi a última avaliação conjunta para RTE em inglês.

Mais recentemente, foi disponibilizado o corpus SNLI (*Stanford Natural Language Inference*) (Bowman et al., 2015), com cerca de 550 mil pares de sentenças anotados para inferência textual, o maior corpus do gênero até o momento. O SNLI não foi utilizado em nenhuma avaliação conjunta, mas diversos artigos têm sido publicados com experimentos sobre o mesmo, focando normalmente em métodos de *deep learning* (Rocktäschel et al., 2015; Wang & Jiang, 2015). O SNLI e o SICK foram criados a partir de descrições de imagens. No SICK, um processo semi-automático gerou uma segunda sentença para cada descrição, introduzindo negações, trocando palavras, entre outras alterações. Já no SNLI, anotadores escreveram, para cada sentença original, três outras: uma que fosse implicada pela primeira, outra que a contradissesse e uma terceira neutra.

A detecção de similaridade semântica textual foi introduzida em 2012 e, em 2013, foi parte do evento *SEM, acontecendo em conjunto com o SemEval (Agirre et al., 2012, 2013). Desde então, a STS tem sido anualmente uma das tarefas propostas no SemEval. Os pares usados nas avaliações de STS incluem sentenças de diferentes origens, como descrições de vídeos e imagens, manchetes de jornais e diferentes traduções de um mesmo texto.

3 Definição das Tarefas

Apresentamos nessa seção os dois fenômenos anotados no corpus.

3.1 Similaridade semântica

Nossos valores para similaridade semântica variam de 1 a 5, como no corpus SICK, de modo que quanto maior o valor, maior a semelhança do significado das duas sentenças. Esse tipo de medida é inerentemente subjetiva, e não conseguimos chegar a uma definição exata para o que cada valor deveria indicar. Ainda assim, as diretrizes gerais para a pontuação utilizadas no ASSIN 2016 seguem abaixo:

1. As sentenças são completamente diferentes. É possível que elas falem do mesmo fato, mas isso não é visível examinando-as isoladamente, sem contexto.
2. As sentenças se referem a fatos diferentes e não são semelhantes entre si, mas são sobre o mesmo assunto (jogo de futebol, votações, variações cambiais, acidentes, lançamento de produtos).
3. As sentenças têm alguma semelhança entre si, e podem se referir ao mesmo fato ou não.
4. O conteúdo das sentenças é muito semelhante, mas uma (ou ambas) tem alguma informação exclusiva. A diferença pode ser

mencionar uma data, local, quantidade diferente, ou mesmo um sujeito ou objeto diferente.

5. As sentenças têm praticamente o mesmo significado, possivelmente com uma diferença mínima (como um adjetivo que não altera a sua interpretação).

A Tabela 1 mostra exemplos de pares em cada um dos níveis. As diretrizes de anotação requisitavam que se considerasse o conteúdo das sentenças em análise, e não os contextos possíveis nos quais elas poderiam aparecer. Por exemplo, considere o exemplo de similaridade 1 na Tabela 1. Embora seja possível que ambas as sentenças venham do mesmo texto e sejam fortemente relacionadas (o que é o caso nesse exemplo), a anotação não deve considerar essas suposições.

3.2 Inferência Textual

Dagan et al. (2013) definem inferência textual como uma relação unidirecional entre um texto (ou premissa) T e uma hipótese H. Se uma pessoa ao ler T conclui que H é verdadeiro, diz-se que T implica (*entails*) H. Embora seja uma definição subjetiva, ela é largamente aceita na comunidade de processamento de línguas naturais, dada a dificuldade de se chegar a uma definição mais precisa.

É comum a distinção entre pares de textos sem inferência e com contradições em conjuntos de dados de inferência textual. Embora seja interessante a distinção, no corpus ASSIN 2016 eles são raros e dessa forma decidimos não criar uma classe separada. Vale lembrar que, tanto no SICK quanto no SNLI (Bowman et al., 2015), pares com contradição são deliberadamente criados, seja manual ou semi-automaticamente.

Nós também definimos uma classe separada para paráfrases, que embora não sejam frequentes, aparecem em nosso corpus de textos jornalísticos. A Tabela 2 mostra um caso em que a primeira sentença implica a segunda; um caso de implicação mútua ou paráfrase; e um terceiro caso em que não há implicação.

4 Criação do Corpus

Nesta seção descrevemos a criação do corpus e apresentamos as estatísticas da anotação.

4.1 Coleta e Anotação do Corpus

A exploração de agrupamentos de notícias para aquisição de pares de sentenças similares não é uma ideia nova; já foi explorada com sucesso em vários trabalhos da literatura (Dolan et al., 2004; Dagan et al., 2005). Entretanto, em vez de anotadores humanos selecionarem pares com base na sobreposição de palavras, empregamos o *Latent Dirichlet Allocation* (LDA) (Blei et al., 2003) para selecionar pares similares.

O LDA, um método de modelagem de espaços vetoriais, atribui uma pontuação para pares de documentos, refletindo o quão similares são entre si. Em um experimento piloto reportado em (Fonseca & Aluísio, 2015), notamos que, em comparação com outros métodos de espaço vetorial, o LDA fornecia os pares mais interessantes para inferência textual, pois recuperava o menor número de sentenças sem relação de inferência (que costumam ser a maioria) e era eficiente em detectar similaridades além da sobreposição de palavras.

Usamos um modelo diferente de LDA para cada variante do português, ambos treinados em grandes corpora de notícias. O modelo para o português do Brasil foi treinado em um corpus coletado do site de notícias G1[2] e o para português europeu com textos do jornal Público[3]. Esses corpora foram somente usados para gerar os modelos LDA, não para coletar os pares de sentenças do corpus ASSIN.

Grupos de notícias sobre o mesmo evento foram coletados do Google News em suas versões específicas para Brasil e Portugal. Filtramos alguns domínios para evitar sites de notícias brasileiros na seção de Portugal e vice-versa. Dados os grupos de notícia coletados e um modelo de espaço vetorial treinado, a criação do nosso corpus seguiu um processo de três etapas:

1. Usamos LDA para encontrar pares de sentenças similares dentro de cada grupo. Esse passo pode ser parametrizado fixando os valores mínimo e máximo de similaridade s_{max} e s_{min}: fixando um valor máximo evita pares de sentenças quase iguais, que seriam classificados trivialmente como paráfrases, e fixando um mínimo evita pares muito dissimilares que são facilmente classificados como sem relação. Fixamos a proporção α de tokens que são encontrados em uma sentença mas não em outra (sem contar *stopwords*). Finalmente, sentenças podem ser limitadas por um tamanho máximo; em uma análise

[2] http://g1.globo.com/
[3] http://www.publico.pt/

1	Mas esta é a primeira vez que um chefe da Igreja Católica usa a palavra em público. A Alemanha reconheceu ontem pela primeira vez o genocídio armênio.
2	Como era esperado, o primeiro tempo foi marcado pelo equilíbrio. No segundo tempo, o panorama da partida não mudou.
3	Houve pelo menos sete mortos, entre os quais um cidadão moçambicano, e 300 pessoas foram detidas. Mais de 300 pessoas foram detidas por participar de atos de vandalismo.
4	A organização criminosa é formada por diversos empresários e por um deputado estadual. Segundo a investigação, diversos empresários e um deputado estadual integram o grupo.
5	Outros 8.869 fizeram a quadra e ganharão R$ 356,43 cada um. Na quadra 8.869 apostadores acertaram, o prêmio é de R$ 356,43 para cada.

Tabela 1: Exemplos para os valores de similaridade semântica.

Inferência	Como não houve acordo, a reunião será retomada nesta terça, a partir das 10h. As partes voltam a se reunir nesta terça, às 10h.
Paráfrase	Vou convocar um congresso extraordinário para me substituir enquanto presidente. Vou organizar um congresso extraordinário para se realizar a minha substituição como presidente.
Sem relação	As apostas podem ser feitas até as 19h (de Brasília). As apostas podem ser feitas em qualquer lotérica do país.

Tabela 2: Exemplos para as categorias de inferência textual.

preliminar, notamos que sentenças muito longas têm muita informação e dificilmente podem ser completamente implicadas por outra.

2. Revisamos os pares coletados em um processo manual. Se um par contém uma sentença sem sentido, é descartado. Sentenças foram também editadas para correção de erros ortográficos e gramaticais, ou para alterar casos em que a presença de implicação é pouco clara.

3. Os pares são anotados. Quatro pessoas anotaram cada par, selecionadas aleatoriamente pelo sistema de anotação. Cada anotador seleciona um valor de similaridade de 1 a 5, e também uma das quatro opções para inferência: a primeira sentença implica a segunda; a segunda implica a primeira; paráfrase, ou nenhuma relação.

Realizamos esse processo em vários lotes, variando os parâmetros. Usamos os valores de s_{min} de 0.65 e 0.6, sem obter grande diferença no resultado. s_{max} foi fixado em 0.9. A proporção de tokens exclusivos para cada sentença foi fixada em 0.1 como mínimo e valores máximos variando entre 0.7 ou 0.8. Com o último valor, notamos um aumento considerável de pares de sentenças com valor de similaridade baixo.

Dada a subjetividade da anotação, definimos algumas diretrizes para lidar com alguns fenômenos linguísticos recorrentes que tinham diferentes interpretações por parte dos anotadores. As diretrizes são voltadas especialmente para a anotação de inferência, e estão listadas na Tabela 3.

Descartamos pares sem concordância de, pelo menos, três votos para a tarefa de inferência textual. Nosso entendimento foi que esses pares eram controversos e assim não seriam boas escolhas para serem incluídos no corpus final. Note-se que os anotadores poderiam indicar implicação tanto da primeira para a segunda sentença como da segunda para a primeira; porém, no corpus final, invertemos a ordem dos pares necessários para que todos os casos de inferência fossem da primeira sentença para a segunda. O valor final de similaridade para cada par é média das quatro pontuações. Dessa forma, os valores são reais separados por intervalos de 0,25.

A anotação foi realizada via uma interface Web construída especialmente para a tarefa, mas flexível o bastante para permitir customizações em futuras anotações. Os anotadores receberam treinamento para calibrar os conceitos das tarefas a serem realizadas, com ajuda de um conjunto de 18 pares exemplificando todos os fenômenos tratados. Em caso de dúvidas, perguntas poderiam ser enviadas via e-mail para a equipe de anotadores, o que permitia discutir casos muito difíceis de decidir, principalmente no começo da anotação.

Por fim, o corpus foi dividido em seções de treinamento (com três mil pares de cada variante) e teste (com os dois mil restantes de cada). A metade brasileira do corpus de treinamento foi disponibilizada em 20 de novembro de 2015, e a metade portuguesa foi disponibilizada dois meses depois.

Conceito	Explicação
Atemporalidade	A interpretação das sentenças não deveria levar em conta a data corrente, de modo que a anotação fizesse sentido no futuro. Assim, embora *há 70 anos atrás* e *em 1945* sejam equivalentes em 2015, devem ser considerados distintos pelos anotadores.
Entidades Nomeadas	Entidades nomeadas que aparecem nas duas sentenças, tendo um aposto ou adjetivo em uma delas, devem ser consideradas equivalentes. *Florianópolis, em Santa Catarina* é equivalente a apenas *Florianópolis*.
Discurso Indireto	Uma sentença com discurso indireto (i.e., *O embaixador disse que (...)*) pode implicar outra que contenha apenas a fala atribuída. O contrário, no entanto, não é possível.
Quantidades	Valores numéricos diferentes só podem ser aceitos para paráfrase/implicação se tiverem indicadores explícitos de serem aproximações: *acerca de*, *pelo menos*, *quase*, *perto de*, etc. Por exemplo, *arrecadou 7 milhões* não implica arrecadou *6 milhões* pois, mesmo sendo uma quantia menor, é possível que se refira a outro evento.

Tabela 3: Resumo das Diretrizes para Anotação.

4.2 Estatísticas da Anotação

O corpus foi anotado por 36 pessoas, que participaram em diferentes quantidades: o anotador com menor participação julgou 25 pares, enquanto o com maior participação julgou 6.740.

Do total de pares anotados, 11.3% foram descartados por não terem três julgamentos iguais quanto à implicação. A proporção é um pouco menor do que as reportadas na criação dos corpora RTE Challenge (Dagan et al., 2005; Giampiccolo et al., 2007). No total, o ASSIN tem 10 mil pares, sendo metade em português brasileiro e metade em português europeu.

A Tabela 4 sumariza estatísticas da anotação. A correlação ρ de Pearson é uma boa métrica para a concordância entre anotadores (ou para o desempenho de um sistema), tendo sido usada também pelos organizadores das competições de STS. Essa medida avalia a dependência linear entre duas variáveis, o que é mais informativo do que apenas a correlação de ranqueamento (computável com a correlação de Spearman). Por exemplo, se um anotador avalia três pares com semelhança 2, 3 e 4, enquanto outro avalia os mesmos com 2, 4 e 5, o ranqueamento é idêntico, mas o valor de ρ está abaixo de 1 por não serem duas variáveis (perfeitamente) linearmente dependentes.

O valor de ρ apresentado na tabela se refere à média das correlações calculadas entre todos os anotadores, ponderada pela quantidade de pares que cada um anotou. Para cada anotador, calculamos a correlação das suas pontuações de similaridade com as médias das pontuações dos pares que ele ou ela anotou (excluindo a sua anotação do cômputo). Para efeito de comparação, a anotação do STS 2015 obteve valores entre 0.65 e 0.85, o que mostra que alcançamos boa concordância entre anotadores quanto à similaridade.

Métrica	Valor
Correlação de Pearson	0,74
Desvio Padrão Médio	0,49
κ de Fleiss	0,61
Concordância	0,80

Tabela 4: Estatísticas da Anotação. Os primeiros 2 valores se referem à anotação de similaridade; os 2 últimos valores à inferência.

O desvio padrão médio avalia a divergência dos julgamentos de similaridade dos pares. É calculado como a média dos desvios padrão de todos os pares no corpus; esses, por sua vez, são calculados como o desvio padrão das quatro pontuações em relação à média do par. O valor reportado na anotação do SICK é de 0,76, indicando que as pontuações dos nossos anotadores divergiram menos.

Com relação à inferência, o valor da concordância κ de Fleiss foi relativamente baixo, o que indica que a anotação desta tarefa de fato envolveu boa quantidade de subjetividade. Os corpora dos desafios RTE, por exemplo, tiveram uma taxa de concordância maior: 0,6 na primeira edição (Dagan et al., 2005), mas chegando a 0,75 ou mais nas subsequentes (Giampiccolo et al., 2007). Entretanto, deve ser notado que esses corpora tratam de sentenças curtas como segundo componente do par (a sentença implicada), o que torna a decisão mais fácil.

A última linha da tabela se refere à concordância simples entre os anotadores. Isso significa que, em 80% dos casos, dois anotadores que julgaram o mesmo par escolheram a mesma categoria de inferência.

As tabelas 5 e 6 mostram estatísticas sobre as anotações de similaridade e inferência, respectivamente. Pode-se ver que as pontuações de si-

milaridade mais comuns estão no intervalo entre 2 e 3. Já quanto à inferência, percebe-se que a relação neutra é a classe majoritária, enquanto as paráfrases são uma porção pequena do corpus.

Similaridade	**PB**	**PE**	**Total**
4,0 – 5,00	1.074	1.336	2.410
3,0 – 3,75	1.591	1.281	2.872
2,0 – 2,75	1.986	1.828	3.814
1,0 – 1,75	349	555	904
Média	3,05	3,05	3,05

Tabela 5: Estatísticas de similaridade do ASSIN.

Relação	**PB**	**PE**	**Total**
Sem relação	3.884	3.432	7.316
Implicação	870	1.210	2.080
Paráfrase	246	358	604

Tabela 6: Estatísticas de inferência do ASSIN.

A pouca quantidade de pares com relação de inferência foi notada já durante nossa análise de um corpus piloto, que não foi incluído no corpus final. Isso se devia ao fato de que, em muitos casos, apenas alguns detalhes impediam que houvesse tal relação: a menção a um local, tempo, propósito, entre outros. Essa situação é ilustrada no exemplo a seguir.

(1) a. O Internacional manteve a boa fase e venceu o Strongest por 1 a 0 nesta quarta-feira, garantindo a liderança do Grupo 4 da Libertadores.

b. Em casa, a equipe gaúcha derrotou o The Strongest, por 1 a 0, e garantiu a primeira colocação do Grupo 4 da Copa Libertadores.

Apesar de as duas sentenças compartilharem a maior parte do conteúdo, cada uma tem alguma informação específica que não é implicada pela outra. A primeira menciona o nome da equipe, além de que estava em boa fase e que o jogo foi na quarta-feira. Já a segunda diz que o jogo foi na casa da equipe, sem explicitar seu nome. Esse tipo de fenômeno é particularmente comum quando se tratam de sentenças longas.

Visando aumentar a proporção de pares com inferência, realizamos pequenas mudanças nas sentenças durante a segunda etapa do nosso processo listado na Seção 4.1. Assim, passamos a remover pequenos trechos de uma ou ambas as sentenças, caso as alterações possibilitassem a inferência. Apesar da proporção final estar menos desequilibrada que o observado em nosso corpus piloto, ainda temos menos pares com inferência e especialmente paráfrases do que o que gostaríamos.

5 Sistemas Participantes

Participaram do ASSIN seis equipes, sendo três brasileiras e três portuguesas. Cada equipe participante pôde enviar o resultado de até três execuções de seus sistemas para cada combinação de variante da língua e subtarefa.

Na tarefa de similaridade, participaram todas as seis equipes inscritas, enquanto quatro participaram da tarefa de inferência textual. A L2F/INESC-ID foi a única a reportar resultados apenas para uma variante; no caso, o português europeu[4].

É interessante notar que os participantes adotaram estratégias bastante diversas entre si, o que permite uma análise de diferentes pontos de vista sobre as tarefas. Ressaltamos também que as equipes que participaram de ambas as tarefas usaram os mesmos atributos para treinar diferentes modelos (em alguns casos, com uma etapa intermediária de seleção automática de atributos).

Portanto, não fazemos aqui uma separação entre abordagens específicas de cada subtarefa; em vez disso, resumimos brevemente o funcionamento dos sistemas dos participantes a seguir.

5.1 Abordagens

A equipe Solo Queue (Hartmann, 2016) utilizou uma abordagem bastante simples, baseada apenas no valor da similaridade do cosseno de duas representações vetoriais de cada sentença. Tais representações são geradas como a soma dos vetores de cada palavra, que por sua vez são obtidas por meio de TF-IDF e word2vec (Mikolov et al., 2013).

Em seguida, os cossenos entre as duas representações (TF-IDF e word2vec) de cada sentença são dadas como entrada para um regressor linear que determina a similaridade do par.

O sistema de L2F/INESC-ID (Fialho et al., 2016) consistiu em extrair diversas métricas dos pares de sentenças, como distância de edição, palavras em comum (incluindo métricas separadas para entidades nomeadas ou verbos modais),

[4]Os autores informaram que não houve tempo o suficiente para treinar os seus modelos para o português do Brasil antes do prazo da avaliação conjunta. Ainda assim, apresentam em seu artigo resultados obtidos após a data.

BLEU, ROUGE etc. Tais métricas foram computadas tanto das sentenças originais como de outras versões, que poderiam estar em caixa baixa, com palavras radicalizadas, usando clusters de palavras (Turian et al., 2010), entre outras modificações. A combinação de diferentes versões das sentenças com as diferentes métricas gerou mais de 90 atributos para descrever cada par, que são então usados para treinar um Kernel Ridge Regression (para similaridade) e um SVM (para inferência).

Fialho et al. (2016) experimentaram ainda aumentar o conjunto de treinamento com uma versão do corpus SICK traduzida automaticamente para o português. No entanto, os resultados obtidos ao se treinar o regressor na versão aumentada foram inferiores, provavelmente devido aos erros de tradução. Por fim, os autores avaliam seus modelos quando treinados em uma variante do português e testados na outra.

As equipes ASAPP e Reciclagem (Alves et al., 2016) compartilharam um módulo de análises de relações lexicais baseado em redes semânticas (como tesauros e wordnets). Diversas métricas baseadas em tais relações foram extraídas dessas redes.

O Reciclagem não conta com nenhum módulo de aprendizado de máquina, empregando apenas métricas de similaridade baseadas nas relações semânticas entre as palavras das duas sentenças. Nesse sentido, o método teve um caráter exploratório quanto à capacidade de diferentes redes semânticas contribuírem para a tarefa de STS e do quanto um sistema sem treinamento poderia alcançar em termos de performance.

Já o ASAPP emprega, além das métricas usadas pelo Reciclagem, atributos como contagem de tokens de cada sentença, orações nominais, tipos de entidades nomeadas etc., todos dados como entrada para classificadores e regressores. Em suas três execuções, foram exploradas formas de partição de dados, combinação de modelos e redução da quantidade de atributos.

Barbosa et al. (2016) utilizaram a estratégia proposta por Kenter & de Rijke (2015): são obtidas representações vetoriais das palavras (no caso, foi usado o word2vec) e, em seguida, os vetores de uma sentença são comparados com os da outra, obtendo-se medidas baseadas no cosseno e a distância euclidiana.

Todas as medidas obtidas são então agrupadas em histogramas, com intervalos pré-definidos. São usados diferentes histogramas para cada medida, e as suas contagens são dados como entrada para os modelos de aprendizado de máquina. Para a tarefa de similaridade, foram usados SVR e o método Lasso, e para a inferência, apenas um SVM.

Também foram explorados métodos baseados em redes neurais recorrentes e convolucionais, usando uma arquitetura siamesa. Esse tipo de arquitetura usa o mesmo conjunto de pesos para mapear cada uma das sentenças para um vetor. Dados os dois vetores, pode ser calculado diretamente o seu cosseno, que é então mapeado para um valor de similaridade. No entanto, a despeito dos bons resultados reportados na literatura recente em PLN, as redes neurais obtiveram resultados muito abaixo dos outros métodos usados pela equipe. A provável causa desta desparidade é a quantidade relativamente pequena de dados disponíveis no ASSIN.

A equipe FlexSTS (Freire et al., 2016) apresentou um framework para calcular a similaridade semântica textual baseada em combinar medidas de semelhança entre tokens de acordo com alinhamentos entre eles. Foram exploradas três configurações: a primeira treina um regressor usando apenas uma função DICE e medidas de distâncias entre os tokens na WordNet. Foi usada a WordNet da língua inglesa, e os pares do ASSIN foram traduzidos automaticamente para consultá-la.

A segunda abordagem do FlexSTS usou apenas o modelo HAL (Hyperspace Analogue to Language) para calcular a similaridade entre as palavras mais similares, enquanto a terceira abordagem combina o modelo HAL com a WordNet. Essas duas não usam nenhum componente de aprendizado de máquina, recorrendo a fórmulas pré-definidas para computar o valor de similaridade de cada par.

6 Avaliação e Resultados

Os participantes receberam o conjunto de teste (sem os rótulos corretos dos pares) em 4 de março de 2016, e tiveram 8 dias para enviar aos organizadores os arquivos com as respostas produzidas por seus sistemas. Cada participante pôde enviar até três resultados.

As métricas usadas na avaliação das duas tarefas são consoantes com as usadas em avaliações conjuntas internacionais. Na tarefa de similaridade textual, foi usada a correlação de Pearson, tendo o erro quadrático médio (MSE, *mean square error*) como medida secundária. Idealmente, os sistemas devem ter a maior correlação possível e o menor MSE possível. Para a inferência, foi usada a medida F1, tendo a acurácia como medida secundária.

6.1 Baselines

Foram usadas duas estratégias como *baseline* para o ASSIN: a primeira memoriza a média das similaridades do corpus de treino e a classe de inferência mais comum, e emite esses valores para todos os pares de teste. A segunda, um pouco mais sofisticada, consiste no treinamento de um classificador baseado em regressão logística e um regressor linear. Estes dois modelos são treinados com apenas dois atributos: a proporção de tokens exclusivos da primeira e da segunda sentença.

6.2 Resultados

As Tabelas 7 e 8 listam os resultados das tarefas de similaridade e inferência, respectivamente, obtidos por cada participante em suas três execuções, bem como os resultados dos sistemas *baseline*.

A equipe Solo Queue (Hartmann, 2016) obteve os melhores resultados da similaridade semântica para o português do Brasil, enquanto o Blue Man Group (Barbosa et al., 2016) obteve os melhores resultados para inferência textual . Já com o português europeu, a L2F/INESC-ID (Fialho et al., 2016) alcançou os melhores resultados nas duas tarefas.

O primeiro *baseline* obteve 0 na correlação de Pearson pelo fato de não haver variação em suas respostas, e a medida ser baseada na correlação de duas variáveis. Ao se combinar as respostas para as duas metades do corpus, é obtido um valor negativo, indicando uma performance pior que dar a mesma resposta sempre.

No entanto, considerando o MSE, esse *baseline* teve resultados melhores que algumas execuções dos participantes, o que significa que tais execuções computaram valores muito distantes da similaridade real dos pares. Já o segundo *baseline* teve resultados competitivos, chegando a superar diversas execuções.

Quanto à inferência, com resultados na Tabela 8, o primeiro *baseline* é também facilmente superado, mas o segundo se saiu bastante bem. Na variante brasileira, chegou a superar todos os três participantes e, na europeia, apenas uma execução da L2F/INESC-ID se saiu melhor.

O último resultado foi bastante inesperado. Apesar de toda a modelagem do problema feita pelas equipes participantes, um *baseline* com apenas dois atributos simples, sem acesso a nenhum recurso externo e usando apenas modelos lineares foi capaz de superar quase todos os sistemas empregados na tarefa. Ao mesmo tempo, esse resultado indica que a presença de inferência no ASSIN é fortemente relacionada com a sobreposição lexical, ainda que tenhamos nos esforçado em incluir tanto pares com inferência que tivessem palavras distintas quanto pares sem relação e palavras compartilhadas.

7 Conclusões

Descrevemos a proposta da Avaliação de Similaridade Semântica e Inferência Textual, como foi criado seu corpus anotado, quais foram as equipes participantes da avaliação conjunta e os resultados que obtiveram. Apresentamos, ainda, dois sistemas *baseline* bastante simples, mas dos quais um superou a maioria dos participantes na tarefa de inferência textual.

O ASSIN 2016 cumpriu seu objetivo de trazer a primeira avaliação conjunta de inferência textual e similaridade semântica para o português. Listamos a seguir algumas conclusões que dizem respeito à criação do corpus e aos sistemas desenvolvidos para a tarefa.

7.1 Criação do Corpus

O método que usamos para a compilação do corpus, apesar de funcional, apresenta alguns empecilhos. O primeiro é o gargalo da etapa de limpeza antes da anotação em si. Durante essa etapa, os critérios para se eliminar ou editar pares são bastante delicados, como nossa experiência mostrou. É uma parte da anotação que deve ficar a cargo de pessoas que tenham conhecimento sobre a tarefa e seus objetivos, e dificilmente poderia ser delegada para uma plataforma de *crowdsourcing*.

Outra dificuldade diz respeito à subjetividade da tarefa. Em alguns casos, os anotadores gastaram bastante tempo tentando se decidir quanto aos julgamentos que deveriam dar para certos pares. Esse tipo de problema retoma o anterior: certas alterações no conteúdo das sentenças torna as decisões mais fáceis, e portanto, a anotação mais confiável e produtiva.

7.2 Sistemas Participantes

Os participantes do ASSIN exploraram diferentes tipos de estratégia para as duas tarefas propostas. É particularmente interessante notar que dentre os melhores resultados obtidos estão duas abordagens muito simples: na similaridade semântica, a comparação da combinação de vetores de palavras, como feito pelo Solo Queue; e para inferência, a comparação da proporção de

		PB		PE		Geral	
Equipe	**Exec.**	**Pearson**	**MSE**	**Pearson**	**MSE**	**Pearson**	**MSE**
Solo Queue	1	0,58	0,50	0,55	0,83	0,56	0,66
	2	0,68	0,41	0,00	1,55	0,29	0,98
	3	**0,70**	**0,38**	0,70	0,66	**0,68**	**0,52**
Reciclagem	1	0,59	1,36	0,54	1,10	0,53	1,23
	2	0,59	1,31	0,53	1,14	0,54	1,23
	3	0,58	1,37	0,53	1,18	0,53	1,27
Blue Man Group	1	0,65	0,44	0,63	0,73	0,63	0,59
	2	0,64	0,45	0,64	0,72	0,63	0,59
ASAPP	1	0,65	0,44	0,68	0,70	0,65	0,57
	2	0,65	0,44	0,67	0,71	0,64	0,58
	3	0,65	0,44	0,68	0,73	0,65	0,58
LEC-UNIFOR	1	0,62	0,47	0,64	0,72	0,62	0,59
	2	0,56	2,83	0,59	2,49	0,57	2,66
	3	0,61	1,29	0,63	1,04	0,61	1,17
L2F/INESC-ID	1			**0,73**	**0,61**		
	2			0,63	0,70		
	3			0,63	0,70		
Baseline (média)	–	0,00	0,76	0,00	1,19	-0,08	0,97
Baseline (sobreposição)	–	0,63	0,46	0,64	0,75	0,62	0,60

Tabela 7: Resultados de todas as execuções para a tarefa de similaridade semântica.

		PB		PE		Geral	
Equipe	**Exec.**	**Acurácia**	**F1**	**Acurácia**	**F1**	**Acurácia**	**F1**
Reciclagem	1	77,65%	0,29	73,10%	0,43	75,38%	0,40
	2	79,05%	0,39	72,10%	0,38	75,58%	0,38
	3	78,30%	0,33	70,80%	0,32	74,55%	0,32
Blue Man Group	2	81,65%	0,52	77,60%	0,61	79,62%	0,58
ASAPP	1	81,20%	0,50	77,75%	0,57	79,47%	0,54
	2	81,65%	0,47	78,90%	0,58	80,27%	0,54
	3	77,10%	0,50	74,35%	0,59	75,72%	0,55
L2F/INESC-ID	1			**83,85%**	**0,70**		
	2			78,50%	0,58		
	3			78,50%	0,58		
Baseline (maioria)	–	77,65%	0,29	69.30%	0,27	73,47%	0,28
Baseline (sobreposição)	–	**82,80%**	**0,64**	81,75%	**0,70**	**82,27%**	**0,67**

Tabela 8: Resultados de todas as execuções para a tarefa de inferência textual.

palavras exclusivas de cada sentença, que foi um dos *baselines* propostos.

Todavia, a equipe L2F/INESC-ID obteve os melhores resultados do ASSIN na variante europeia (a única em que competiu), empregando um sistema baseado em um rico conjunto de atributos. Esse resultado indica que superar métodos simples como os listados acima requer uma modelagem extensiva do problema.

Outra linha de pesquisa bastante bem sucedida na literatura recente são redes neurais recorrentes (como LSTMs) ou convolucionais. O Blue Man Group foi o único grupo a explorá-las, mas as descartou após obter resultados preliminares negativos. Uma possível explicação para esse fato é que o conjunto de dados do ASSIN é menor e com sentenças mais complexas do que as que se encontram para conjuntos semelhantes

em inglês, onde os modelos neurais obtêm os melhores resultados.

Por fim, notamos que nenhum dos participantes modelou as sentenças em alguma estrutura sintática ou semântica; em vez disso, todos exploraram apenas o nível lexical. Pelo menos para a inferência textual, há evidências na literatura de que a compreensão da estrutura das sentenças tem um papel importante (Dagan et al., 2013), e a ausência desse tipo de análise pode explicar o desempenho dos sistemas abaixo do *baseline*.

7.3 Trabalhos Futuros

Novas edições do ASSIN teriam o potencial de estimular e melhorar a pesquisa nas duas tarefas propostas para a língua portuguesa. No entanto, acreditamos que seria interessante trabalhar com outros tipos de pares de sentença, especialmente na tarefa de inferência.

Uma possibilidade seria o uso de pares de sentenças escritos especificamente com o objetivo de terem ou não uma relação de implicação, como foi feito no SICK e SNLI. Nesse caso, a subjetividade da anotação é reduzida drasticamente, com o preço de não se trabalhar com um cenário realista. De fato, a motivação principal da criação destes dois corpora foi fornecer um ambiente para sistemas de PLN aprenderem o funcionamento de certos mecanismos da linguagem humana.

Outro direcionamento seria usar apenas fatos simples, na forma de sentenças com uma única oração, como o segundo componente de cada par. Essa foi a estratégia adotada na criação dos corpora dos RTE Challenges, e mantêm o realismo da tarefa na medida em que a primeira sentença pode ser extraída de um jornal ou outra fonte real. Por outro lado, esse cenário não requer que os sistemas processem e comparem duas sentenças inteiras, mas apenas busque por confirmação de um fato.

Por fim, uma estratégia que facilitasse a anotação do corpus seria também interessante por permitir a criação um novo recurso em maior escala, tornando mais viável a exploração de métodos neurais que necessitam de grandes bases de treinamento.

Agradecimentos

Agradecemos o apoio da Fapesp, processos número 2016/02466-5 e 2013/22973-0, o apoio do CNPq, processos número 155137/2015-8 e 153047/2016-0, e também o apoio da Google via programa *Google Research Awards for Latin America*, projeto 23327 Google/FUNDEP Google Research Grant para o desenvolvimento dessa pesquisa.

Referências

Agirre, Eneko, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo 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 International Workshop on Semantic Evaluation (SemEval 2015)*, 252–263.

Agirre, Eneko, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre & Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. Em **SEM 2013: The Second Joint Conference on Lexical and Computational Semantics.*, 32–43. Association for Computational Linguistics.

Agirre, Eneko, Daniel M. Cer, Mona T. Diab & Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. Em *Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2012, Montréal, Canada, June 7-8, 2012*, 385–393.

Alves, Ana Oliveira, Ricardo Rodrigues & Hugo Gonçalo Oliveira. 2016. ASAPP: alinhamento semântico automático de palavras aplicado ao português. *Linguamática* 8(2). 43–58.

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

Bentivogli, Luisa, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo & Bernardo Magnini. 2009. The fifth Pascal recognizing textual entailment challenge. Em *Proceedings of the Text Analysis Conference 2009*, s.pp.

Blei, David M., Andrew Y. Ng & Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3. 993–1022.

Bowman, Samuel R., Gabor Angeli, Christopher Potts & Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. Em *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 632–642. ACL.

Dagan, Ido, Oren Glickman & Bernardo Magnini. 2005. The PASCAL recognising textual entailment challenge. Em *Proceedings of the PASCAL challenges on Recognizing Textual Entailment*, 177–190.

Dagan, Ido, Dan Roth, Mark Sammons & Fabio Massimo Zanzotto. 2013. *Recognizing Textual Entailment: Models and Applications* Synthesis Lectures on Human Language Technologies. Morgan & Claypool.

Dolan, Bill, Chris Quirk & Chris Brockett. 2004. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. Em *Proceedings of the 20th International Conference on Computational Linguistics*, 350–356.

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 R. & Sandra M. Aluísio. 2015. Semi-Automatic Construction of a Textual Entailment Dataset: Selecting Candidates with Vector Space Models. Em *Proceedings of STIL 2015*, 201–210.

Freire, Jânio, Vládia Pinheiro & David Feitosa. 2016. FlexSTS: Um framework para similaridade semântica textual. *Linguamática* 8(2). 23–31.

Giampiccolo, Danilo, Hoa Trang Dang, Bernardo Magnini, Ido Dagan, Elena Cabrio & Bill Dolan. 2008. The fourth PASCAL recognizing textual entailment challenge. Em *Proceedings of the First Text Analysis Conference*, 1–9.

Giampiccolo, Danilo, Bernardo Magnini, Ido Dagan & Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. Em *Proceedings of the Workshop on Textual Entailment and Paraphrasing*, 1–9.

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

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

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

Mikolov, Tomas, Kai Chen, eg Corrado & Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. Available from arXiv:1301.3781.

Rocktäschel, Tim, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiský & Phil Blunsom. 2015. Reasoning about entailment with neural attention. Available from arXiv:1509.06664.

Turian, Joseph, Lev Ratinov & Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. Em *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–394.

Turney, Peter D. & Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research* 37. 141–188.

Wang, Shuohang & Jing Jiang. 2015. Learning natural language inference with LSTM. Available from arXiv:1512.08849.

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

Blue Man Group no ASSIN: Usando Representações Distribuídas para Similaridade Semântica e Inferência Textual

Blue Man Group at ASSIN:
Using Distributed Representations for Semantic Similarity and Entailment Recognition

Luciano Barbosa
IBM Research
lucianoa@br.ibm.com

Paulo Cavalin
IBM Research
pcavalin@br.ibm.com

Victor Guimarães
IBM Research
victorl@br.ibm.com

Matthias Kormaksson
IBM Research
matkorm@br.ibm.com

Resumo

Neste artigo apresentamos a metodologia e os resultados obtidos pela equipe Blue Man Group, na competição de Avaliação de Similaridade Semântica e Inferência Textual do PROPOR 2016.[1]

A estratégia da equipe consistiu em avaliar métodos baseados no uso de vetores semânticos de palavras, com duas frentes básicas: 1) uso de vetores de características de pequena dimensão, e 2) estratégias de deep learning para vectores de características de grandes dimensões. Os resultados nas bases de avaliação demonstraram que a primeira frente seria mais promissora, e os resultados submetidos para a competição da segunda frente foram descartados.

Com isso, considerando o melhor resultado de cada uma das seis equipes, conseguimos atingir os melhores resultados de acurácia e medida F1 na tarefa de inferência textual, na base de português brasileiro, e o melhor resultado geral de F1 considerando também a base de português de Portugal. Na tarefa de similaridade semântica, a equipe atingiu o segundo lugar na base de português brasileiro, e terceiro lugar considerando ambas as bases.

Palavras chave

Similaridade Semântica, Inferência Textual, Deep Learning, Vetores Semânticos de Palavras

Abstract

In this paper, we present the methodology and the results obtained by our team, dubbed Blue Man Group, in the ASSIN (from the Portuguese *Avaliação de Similaridade Semântica e Inferência Textual*) competition, held at PROPOR 2016.

Our team's strategy consisted of evaluating methods based on semantic word vectors, following two distinct directions: 1) to make use of low-dimensional, compact, feature sets, and 2) deep learning-based strategies dealing with high-dimensional feature vectors. Evaluation results demonstrated that the first strategy was more promising, so that the results from the second strategy have been discarded.

As a result, by considering the best run of each of the six participant teams, we have been able to achieve the best accuracy and F1 values in entailment recognition, in the Brazilian Portuguese set, and the best F1 score considering also the Portuguse from Portugal set. In the semantic similarity task, our team was ranked second in the Brazilian Portuguese set, and third considering both sets.

Keywords

Semantic Similarity, Entailment Recognition, Deep Learning, Word Vectors

1 Introdução

Neste trabalho, apresentamos a metodologia e resultados obtidos pela nossa equipe, nomeada *Blue Man group*, na competição intitulada *Avaliação de Similaridade e Inferência Textual* (ASSIN), a qual foi juntamente realizado com o congresso PROPOR (International Conference on the Computational Processing of Portuguese) em 2016.

A competição ASSIN atribuíu duas tarefas para os participantes: avaliação da similaridade semântica, e reconhecimento de inferência textual. Dadas as sentenças s_1 e s_2, a primeira tarefa consiste em atribuir um valor, representando o grau de relação semântica entre s_1 e s_2. A se-

[1] International Conference on the Computational Processing of the Portuguese Language (http://propor2016.di.fc.ul.pt/)

gunda tarefa envolve determinar se s_1 implica s_2 (a sentença s_1 implica a sentença s_2 se, depois de ler ambas e sabendo que s_1 é verdade, é possível concluir que s_2 também é verdade). Dadas estas duas tarefas, os pesquisadores foram convidados a formar equipes e participar na competição com o desenvolvimento de sistemas para resolver uma ou ambas as tarefas, fazendo uso de dados rotulados fornecidos pela organização da competição, e enviar os seus resultados em um teste cego, ou seja, em dados sem o conhecimento da rotulagem. Vale ressaltar que textos tanto em português do Brasil como em português de Portugal estavam disponíveis, aqui denotados PT-BR e PT-PT, respectivamente, e as equipes podiam optar por apresentar resultados para apenas um ou ambas as variações do português.

Nossa equipe (Blue Man Group) focou em abordagens baseadas em vetores semânticos de palavras (do inglês *word vectors* ou *word embeddings*) para resolver as duas tarefas (maiores detalhes são apresentados na Seção 3). Considerando vetores semânticos de palavras criados com toda a Wikipedia em língua portuguesa, seguimos duas frentes distintas. Na primeira, implementamos um conjunto de características da literatura, proposto por Kenter & de Rijke (2015), para treinar tanto modelos de regressão e classificação baseados em vetores de suporte (do inglês *support vectors*), assim como o modelo de regressão Lasso (do inglês *least absolute shrinkage and selection operator*) (Tibshirani, 1996). Na segunda frente, exploramos métodos de aprendizagem profunda (do inglês *deep learning*) tais quais redes neurais siamesas (do inglês *siamese networks*) (Chopra et al., 2005). As avaliações preliminares com os conjuntos de dados de treinamento e experimentação demonstrou que a primeira direção era mais promissora, fazendo com que decidíssemos por apresentar apenas os resultados da primeira estratégia.

No total, seis equipes participaram da competição. Considerando apenas o melhor resultado de cada equipe, os resultados demonstram que nosso sistema funcionou melhor na tarefa de reconhecimento de inferência textual, já que conquistou o primeiro lugar em acurácia e F1 para o conjunto PT-BR, e o segundo lugar na acurácia e primeiro lugar em F1 na avaliação geral. Na tarefa de avaliação similaridade semântica, os nossos melhores resultados foram o segundo lugar tanto em correlação de Pearson como em Erro Quadrático Médio (MSE) para o conjunto PT-BR, e segundo lugar em Pearson e terceiro em MSE na avaliação geral. Para o conjunto PT-PT, o sistema obteve um desempenho melhor para o reconhecimento de inferência textual, alcançando o segundo melhor valor de F1, mas ficou apenas em quarto lugar na outra tarefa.

No restante deste documento, apresentamos com mais detalhes como o nosso sistema foi desenvolvido e avaliado.

2 Competição ASSIN

Tal como já referido, a competição ASSIN consistiu em um fórum de avaliação para duas tarefas, a similaridade semântica e o reconhecimento de inferência textual, para o qual participantes (ou equipes) poderiam desenvolver sistemas e apresentar os seus resultados nos dados fornecidos pela comissão organizadora. Um grande conjunto de dados contendo pares de sentenças, nas variações de português tanto do Brasil como de Portugal, foi criado para permitir que os participantes desenvolvessem e avaliassem os sistemas. Os participantes poderiam enviar os resultados para uma ou ambas as tarefas, e também para uma ou ambas as variações de português. Em seguida, as equipes seriam classificadas pelos resultados de seus sistemas considerando uma avaliação em outro conjunto de dados, isto é, o conjunto de testes. Tanto as métricas e os conjuntos de dados, assim como as tarefas em questão, são explicadas em detalhes no restante desta seção.

O conjunto de dados ASSIN, contendo um total de 10.000 pares de frases, pode ser dividido nos seguintes subconjuntos. O conjunto de treinamento PT-BR contém 3.000 pares rotulados de frases coletadas do sítio Google News, apenas de fontes brasileiras. O conjunto de treinamento PT-PT também contém 3.000 pares rotulados de frases coletadas do Google News, porém apenas de fontes portuguesas neste caso. E os conjuntos de testes cegos PT-BR e PT-PT, contêm 2.000 pares não rotulados de sentenças cada um, das mesmas fontes utilizadas para os dados de treinamento. Vale ressaltar que as etiquetas dos conjuntos de teste foram disponibilizados para os participantes apenas depois que as equipes apresentaram os seus resultados.

Para a primeira tarefa, isto é, avaliação de similaridade semântica, a similaridade é medida numa escala entre 1 e 5, onde 1 representa que as sentenças são completamente diferentes e 5 representa sentenças com essencialmente o mesmo significado. Assim sendo, as escalas são variações graduais destes dois conceitos. Neste contexto, esta tarefa consiste na construção de um modelo que, dado o par de sentenças $p(i) = (s_1(i), s_2(i))$, contendo a sentenças $s_1(i)$ e $s_2(i)$, prediz o valor de similaridade semântica $y(i)$. Dados os valores

de similaridade $x(i)$ definidos manualmente, os sistemas são avaliadas por meio da correlação de Pearson entre o conjunto que contém todos $x(i)$ e $y(i)$, e o erro quadrático médio (do inglês *mean squared error* - MSE).

A segunda tarefa — reconhecimento de inferência textual (RTE) — consiste em determinar se o significado da hipótese está implicado no texto (Bentivogli et al., 2011). Ou seja, suponha s_1 é o texto e s_2 é a hipótese, s_1 implica s_2 se, após a leitura de ambos e sabendo que s_1 é verdade, uma pessoa concluiu que s_2 também deve ser verdade. Dado que o conjunto de dados fornecido pelo ASSIN também distingue casos de vinculação bidirecional, ou paráfrases, o par de frases s_1 e s_2 devem ser classificados em uma das seguintes classes: *Inferência Textual*, *Paráfrase* e *Nenhuma Relação*. Considerando as etiquetas definidas por inspeção manual, os sistemas são medidos com as medidas denotadas acurácia e pontuação F1.

3 Metodologia

Como já mencionado, a estratégia empregada pela nossa equipe consistiu em avaliar abordagens baseadas em vetores de palavras, onde estes representam o significado semântico das palavras (ver Seção 3.1). Como consequência, duas estratégias distintas foram seguidas. A primeira, apresentada na Seção 3.2, consistiu em implementar um conjunto de características proposto na literatura para representar a semelhança entre os pares de sentenças, para o uso de modelos de regressão como a regressão de vetores de suporte (*support vector regression*, SVR) para avaliação de similaridade semântica, e máquinas de vetor de suporte (*support vector machines*, SVM) para o reconhecimento de inferência textual. E a segunda estratégia, apresentada na Seção 3.3, explorou redes neurais siamesas de aprendizado profundo, com o objetivo de aprender a melhor representação a partir dos dados brutos, ou seja, diretamente a partir dos vectores de palavras dos pares de sentenças.

3.1 Vetores de palavras

Vetores de palavras (do inglês *word vectors* ou *word embeddings*) têm sido utilizados com sucesso ao longo dos últimos anos para aprender representações úteis de palavras, as quais codificam o significado semântico das palavras por meio de vetores contínuos (Collobert et al., 2011). Em outras palavras, mesmo que duas palavras sejam lexicamente escritas de maneiras totalmente distintas, se estas duas palavras apresentarem significados semânticos semelhantes, seus vetores de palavra correspondentes devem ser muito similares. Estes vectores tornam possível não apenas a criação de método de PLN que são capazes de codificar de maneira mais precisa o significado semântico das palavras do vocabulário comparado com o uso apenas de suas formas lexicais, mas estes métodos também permitem tirar proveito de grandes conjuntos de texto sem que haja a necessidade de alguma forma de rotulagem. Os vetores de palavra podem ser criados de maneiras totalmente não-supervisionada.

A aprendizagem de vetores de palavras é feita da seguinte maneira. Dado um grande conjunto de textos, os vetores de palavra são aprendidos ao se considerar a frequência de distribuição de palavras. Isto é, dada uma palavra e as suas palavras anteriores e posteriores em uma frase, um modelo de aprendizagem de máquina tal qual uma rede neural pode ser aprendido, usando as palavras vizinhas como entrada, e a palavra central como saída.

Neste trabalho, os vetores de palavras foram criados com a ferramenta *word2vec*,[2] utilizando como entrada todos os textos em português disponíveis na Wikipédia. Este conjunto contém um total de 636,597 linhas de texto, com 229,658,430 ocorrências de palavras, e um vocabulário com um total 540.638 palavras distintas. A ferramenta *word2vec* foi configurada com os seguintes parâmetros: modelo *skip n-gram*; tamanho de vetor de palavra igual a 300; comprimento máximo de salto entre as palavras definido como 5; 10 exemplos negativos; softmax hierárquica não usada; limiar de ocorrência de palavras estabelecidas para 10^{-4}; e 15 iterações de treinamento.

3.2 Estratégia 1: Características de Kenter e Rijke

3.2.1 Conjunto de características

O conjunto de características proposto por Kenter & de Rijke (2015), consiste em extrair um único vetor de características, denotado $\bar{x}_i = x_{i1}, \ldots, x_{iK}$, para codificar a similaridade semântica do par de sentenças $s_1(i)$ e $s_2(i)$. Neste trabalho, propomos o uso de tal conjunto de características para ambas as tarefas da competição, ou seja, para a avaliação de similaridade semântica e reconhecimento de inferência textual.

Dados os conjuntos de vetores de palavra $\Omega_{i,1}$

[2] http://code.google.com/archive/p/word2vec/

e $\Omega_{i,2}$, calculados a partir das sentenças $s_{i,1}$ e $s_{i,2}$, este conjunto de características é composto por dois tipos de atributos: 1) atributos baseados em redes semânticas; e 2) atributos de nível textual.

Em suma, redes semânticas consistem em construir uma rede (ou grafo) considerando as distâncias dos pares de vetores de palavra $(\omega_{1,j}, \omega_{2,k})$ relacionados a $s_{i,1}$ e $s_{i,2}$, onde

$$\omega_{1,j} \in \Omega_{i,1} \text{ e } \omega_{2,j} \in \Omega_{i,2}.$$

Nesse caso, dois tipos de redes são construídas. O primeiro, denominado rede semântica ponderada por saliência, combina a frequência inversa em documentos (do inglês *inverse document frequency* - IDF) para definir as conexões entre os nós, ao considerar, para cada vetor de palavra $\omega_{1,j}$ pertencente a $\Omega_{i,1}$, o vetor de palavra $\omega_{2,k}$ pertencente a $\Omega_{i,2}$ que é o mais similar àquele vetor, isto é, o vetor de palavra $\omega_{2,k}$ com a menor distância cosseno para $\omega_{1,j}$. Os links na rede ponderada representam as distâncias entre os vetores de palavra correspondentes, multiplicadas pelo IDF do termo correspondente em $s_{i,1}$. Neste trabalho, o IDF é computado no mesmo conjunto usado para criar o conjunto de vetores de palavras, isto é, a Wikipedia português. O segundo tipo de rede, ao qual nos referimos como rede semântica não ponderada, apresenta uma ideia similar à rede já descrita, porém, não se baseia no uso dos IDFs. Neste caso, duas redes não ponderadas são criadas. Uma contém as distâncias entre todos os pares de termos $(\omega_{1,j}, \omega_{2,k})$. E a outra contém as distâncias apenas dos pares $(\omega_{1,j}, \omega_{2,k})$, com menor distância entre si, assim como é feito com as redes semânticas ponderadas por saliência.

No final, as informações nas redes semânticas descritas no parágrafo anterior são usadas para criar histogramas, os quais são concatenadas para compor um único vetor de características. Os limites para estes histogramas foram definidos da seguinte maneira. Para o características calculadas a partir da rede semântica ponderadas por saliência, os valores são 0–$0,15$; $0,15$–$0,4$ e $0,4$–∞. Para ambas as redes semânticas não ponderadas, os valores são -1–$0,45$; $0,45$–$0,8$ e $0,8$–∞.

Além disso, o conjunto de características também inclui atributos de nível textual. Estes atributos são definido de duas formas:

1. a distância entre os vetores de palavra, onde tanto o cosseno e distâncias euclidianas são computados entre os vetores palavra médios de $s_{i,1}$ e $s_{i,2}$;
2. histograma dos valores das dimensões, onde um histograma é calculado a partir dos valores reais apresentados pelos vetores de palavra médios do par de sentenças. Neste caso, os limites para o histograma foram definidos como $-\infty$–$0,001$; $0,001$–$0,01$; $0,01$–$0,02$ e $0,02$–∞.

O conjunto de características resultante é consequentemente composto por um vetor de 15 posições, que correspondem a: 3 características de histograma de redes semânticas ponderados por saliência, 2×3 a partir dos histogramas das duas redes semânticas não ponderadas, 2 baseados nas distâncias dos vetores de palavra médios, e 4 a partir do histograma dos valores das dimensões.

Além disso, vale a pena mencionar que estas 15 características podem ser replicadas através do uso de outros conjuntos de vetores de palavras. Em outras palavras, para cada conjunto distinto de vetores de palavra, um novo vetor de características com 15 posições pode ser extraído. E estes vetores de característica podem ser combinados, por exemplo, a partir da concatenação dos vetores. Neste trabalho, no entanto, consideramos apenas um único conjunto de vetores de palavra, isto é, aquele descrito na Seção 3.1, por questão de simplicidade.

Os detalhes sobre estas características, assim como informação sobre como foram definidos os limites dos histogramas, seguiram a proposta de Kenter & de Rijke (2015).

3.2.2 Regressão e Classificação Baseada em Vetores de Suporte

Máquinas de vetores de suporte (do inglês *Support vector machines* - SVM), e o seu método correspondente para problemas de regressão, isto é, regressão com vetores de suporte (do inglês *Support Vector Regression* - SVR), tornaram-se muito populares nos últimos anos, dado o bom desempenho em um grande número de tarefas (Byun & Lee, 2002). SVM e SVR empregam a seguinte ideia: os vetores de entrada, denotados $x_{i1}, \ldots, x_{iK}$, são não-linearmente mapeados para um espaço de características de muito alta dimensão. Neste espaço de características, uma superfície de decisão não linear é construída, com o intuito de se prever o valor de classe $y_i \in [-1, 1]$, no caso de classificação, ou o valor real y_i, no caso de regressão. Propriedades especiais da superfície de decisão garantem a alta capacidade de generalização dessas máquinas de aprendizagem (Cortes & Vapnik, 1995).

Para este trabalho, ambos SVR e SVM foram implementadas com a biblioteca *Scikit Le-*

arn[3]. Para ambas abordagens, utilizou-se o núcleo Gaussiano após algumas experimentações preliminares. E os parâmetros de configuração de foram configurados por meio de uma busca em grid com validação cruzada, baseada em 5 repartições, usando o conjunto de treinamento.

3.2.3 Lasso

Seja y_i o valor ser predito e $x_{i1}, \ldots, x_{iK}$ denotam as K características calculadas para cada observação i. Considerou-se o seguinte modelo de regressão:

$$y_i = \beta_0 + \sum_{k=1}^{K} \beta_k x_{ik} + \sum_{\ell \neq k} \alpha_{\ell k} x_{i\ell} x_{ik} + \varepsilon_i,$$

onde ε_i denota o erro associado com a observação i. O modelo acima é linear nas características e inclui todas as interações bidirecionais possíveis, $x_{i\ell}x_{ik}$, entre pares de características. Considerando que θ denote o conjunto de todos os parâmetros $(\beta_k)_k$ e $(\alpha_{\ell k})_{\ell k}$. Ao especificar corretamente uma matriz de design X (cujas colunas são as características e correspondente interações bidirecionais), podemos formular a regressão acima em uma notação de matriz mais simples:

$$y = X\theta + \epsilon,$$

onde y e ε são os valores preditos e o vetor de erro, respectivamente.

Observe que, se tivéssemos de estimar o modelo acima, utilizando o método dos mínimos quadrados poderíamos facilmente ter problemas com *over-fitting* devido à grande quantidade de parâmetros a serem estimados:

$$n_{param} = K + 1 + \frac{(K-1)\cdot K}{2} \sim O(K^2).$$

A regressão Lasso (Tibshirani, 1996) foi projetada para lidar com este problema em potencial de *over-fitting*, e pertence a uma classe de modelos chamados de regressão regularizada. Através da aplicação de mínimos quadrados com uma restrição L_1 adicional sobre os parâmetros,

$$\|\theta\|_1 = \sum_k |\theta_k| \leq C,$$

para algum $C > 0$, somos capazes de evitar o *over-fitting*. Este método tem a vantagem de servir como um método de seleção de variáveis, assim como, uma vez que a penalidade L_1 obriga efetivamente que algumas das estimativas dos parâmetros sejam exatamente igual a 0.

[3] http://scikit-learn.org

3.3 Estratégia 2: Redes Siamesas

Redes siamesas (Chopra et al., 2005) têm sido amplamente utilizadas no processamento de imagens e textos, como o objetivo de aprender uma métrica de similaridade de dados. Para a tarefa específica proposta no ASSIN, utilizamos redes siamesas para aprender a semelhança entre duas sentenças em português. Essencialmente, dado um par de sentenças, uma rede siamesa projeta cada frase em um novo espaço de representação, utilizando, por exemplo, redes convolucionais ou recorrentes. Os parâmetros W de cada projeção de sentença são compartilhados. Estas representações são então dadas como entrada para uma métrica de similaridade pré-definida, tal qual as distâncias cosseno ou Euclidiana que calculam a semelhança entre as duas representações. Durante o treinamento, a rede aprende a matriz de parâmetros (W) que minimiza uma dada função de perda. Em nossos experimentos, utilizamos o erro quadrático médio como a função de perda. O erro é a diferença entre o verdadeiro valor de semelhança dada nos dados de treino e o previsto. A partir deste quadro, tentamos diferentes configurações. Por exemplo, para projetar as frases tentamos o uso de redes convolutivas (CNN) (Collobert et al., 2011) e um tipo de redes recorrentes chamada de rede de memória a longo-curto prazo (do inglês *Long-Short Term Memory* - LSTM) (Hochreiter & Schmidhuber, 1997). Usamos similaridade cosseno como a medida de similaridade. E para implementar as redes, usamos a plataforma Keras (Chollet, 2015).

Como mostramos na Seção 4, estas diferentes configurações de redes siamesas não resultaram em bom desempenho no conjunto de dados de teste. Por essa razão, nós não apresentamos os seus resultados para a competição ASSIN.

4 Resultados de Avaliação

Nesta seção, discutimos os resultados obtidos com os métodos descritos no Seção 3. Para tal avaliação, consideramos o conjunto de dados Trial como conjunto de teste, e ambos os conjuntos de treinamento PT-BR e PT-PT. É importante comentar que, no conjunto de treino PT-BR, fizemos a remoção de todas as amostras que também aparecem no conjunto Trial, já que percebemos tal duplicação.

Uma comparação dos resultados para cada método é apresentada na Tabela 1. Neste caso, os melhores resultados foram alcançados com características de Kenter e Rijke tanto com SVRs ou

Configuração	**Similaridade**	**RTE**
Baseline: Bag of Words Geral	0.47	
Características de Kenter e Rijke - SVR(M) PT-BR	0.51	79.60/0.45
Características de Kenter e Rijke - SVR(M) PT-PT	0.49	74.20/0.50
Características de Kenter e Rijke - SVR(M) Geral	0.50	77.00/0.51
Características de Kenter e Rijke - Lasso PT-BR	0.52	
Características de Kenter e Rijke - Lasso PT-PT	0.50	
Características de Kenter e Rijke - Lasso Geral	0.52	
CNN - PT-BR	0.35	
LSTM - PT-BR	0.41	

Tabela 1: Resultados de avaliação (correlação de Pearson), considerando conjunto Trial como conjunto de teste.

	PT-BR				**PT-PT**				**Geral**			
	Sim		**RTE**		**Sim**		**RTE**		**Sim**		**RTE**	
Equipe	**P**	**MSE**	**Acc**	**F1**	**P**	**MSE**	**Acc**	**F1**	**P**	**MSE**	**Acc**	**F1**
Solo Queue	0.70	0.38	-	-	0.70	0.66	-	-	0.68	0.52	-	-
Reciclagem	0.59	1.31	79.05	0.39	0.54	1.10	73.10	0.43	0.54	1.23	75.58	0.40
ASAPP	0.65	0.44	81.65	0.47	0.68	0.70	78.90	0.58	0.65	0.58	80.23	0.54
LEC-UNIFOR	0.62	0.47	-	-	0.64	0.72	-	-	0.62	0.59	-	-
L2F/INESC-ID	-	-	-	-	0.73	0.61	83.85	0.70	-	-	-	-
Blue Man Group	0.65	0.44	81.65	0.52	0.64	0.72	77.60	0.61	0.63	0.59	79.62	0.58

Tabela 2: Os melhores resultados de cada time na competição (Sim: tarefa de avaliação de similaridade semântca; RTE: tarefa de reconhecimento de inferência textual; Acc: acurácia; F1: medida F1; MSE: erro médio quadrático).

Lasso para a avaliação similaridade semântica, e com SVMs para o reconhecimento inferência textual. Com SVR, correlação de Pearson de 0,51, 0,49, e 0,50 foram atingidos nos conjuntos PT-BR, PT-PT, e no geral, respectivamente. Na tarefa de reconhecimento de reconhecimento de inferência textual, as pontuações F1 de 0,45, 0,50, e 0,51, foram alcançados nos mesmos conjuntos, respectivamente. Além disso, observa-se que com Lasso, os resultados são muito semelhantes para aqueles do SVR.

A segunda estratégia, recorrendo às redes siamesas, não alcançou bons resultados. No melhor resultado, a rede LSTM obteu correlação de Pearson de 0,41 usando PT-BR como dados de treinamento, o qual é 0,11 pontos abaixo da nossa melhor estratégia. Por esta razão, decidimos por apresentar apenas os resultados com as características de Kenter, enviando os resultados tanto de SVR e Lasso para a similaridade semântica, e os resultados com SVM para o reconhecimento de inferência textual.

5 Resultados da Competição

Nesta seção, vamos discutir os resultados dos nossos melhores métodos nos dados do teste cego, ou seja, os dados não rotulados de teste, e como foi o desempenho destes métodos comparado aos métodos dos outros concorrentes.

No total, seis equipes participaram da competição. Além de nossa equipe, apenas duas outras equipes apresentaram resultados para ambas as tarefas e para ambos conjuntos PT-BR e PT-PT. Das três equipes restantes, duas focaram apenas na tarefa de similaridade semântica, considerando ambos os conjuntos, e a outra equipe apenas no conjunto PT-PT, nas duas tarefas.

O melhor resultado de cada equipe,[4] ou seja, a melhor tentativa, é apresentado na Tabela 2, e o ranking de cada equipe, também considerando apenas a melhor tentativa, é apresentada na Tabela 3. Considerando apenas a melhor tentativa de cada equipe, conseguimos alcançar resultados muito bons com o conjuntos PT-BR e geral, porém resultados distantes do primeiro lugar no conjunto PT-PT. Com PT-BR, ficamos classificados em primeiro lugar tanto em acurácia como F1 para o reconhecimento de inferência textual, e segundo lugar em similaridade semântica. Além dos bons resultados, foi surpreendente que as características de Kenter apresentaram desempenho melhor em reconhecimento de inferência textual do que na avaliação de simi-

[4]Para cada equipe, foi permitido o envio de até três tentativas diferentes.

	PT-BR				PT-PT				Geral			
	Sim		RTE		Sim		RTE		Sim		RTE	
Equipe	**P**	**MSE**	**Acc**	**F1**	**P**	**MSE**	**Acc**	**F1**	**P**	**MSE**	**Acc**	**F1**
Solo Queue	1st	1st	-	-	2nd	2nd	-	-	1st	1st	-	-
Reciclagem	5th	5th	3rd	3rd	6th	6th	4th	4th	5th	5th	3rd	3rd
ASAPP	2nd	2nd	1st	2nd	3rd	3rd	2nd	3rd	2nd	2nd	1st	2nd
LEC-UNIFOR	4th	4th	-	-	4th	4th	-	-	4th	3rd	-	-
L2F/INESC-ID	-	-	-	-	1st	1st	1st	1st	-	-	-	-
Blue Man Group	2nd	2nd	1st	1st	4th	4th	3rd	2nd	2nd	3rd	2nd	1st

Tabela 3: Posição das equipes considerando a melhor abordagem em cada tarefa e conjunto (Sim: tarefa de avaliação de similaridade semântca; RTE: tarefa de reconhecimento de inferência textual; Acc: acurácia; F1: medida F1; MSE: erro médio quadrático).

laridade semântica, uma vez que o conjunto de características foi originalmente proposto para a última tarefa. No geral, ficamos em primeiro lugar em reconhecimento de inferência textual considerando F1, e em segundo lugar em acurácia. Na similaridade semântica, nossa equipe apresentou o segundo melhor valor de correlação de Pearson e o terceiro melhor valor de MSE. No conjunto PT-PT, conseguimos nos classificar em segundo lugar em F1 para a inferência textual, e terceiro em acurácia. Entretanto, para a similaridade semântica, apenas o quarto lugar (empatado com outra equipe) foi atingido.

Uma observação importante, é que em algumas tarefas ou conjuntos as equipes que alcançaram os melhores resultados foram aquelas que focaram apenas numa tarefa ou conjunto específico. Por exemplo, a equipe *Solo Queue* apresentou resultados apenas para a similaridade semântica, e eles venceram esta tarefa tanto para PT-BR quanto geral, e ficaram em segundo lugar para PT-PT. A equipe *L2F/INESC-ID*, em contrapartida, apresentou resultados apenas para PT-PT, para ambas as tarefas, e obtiveram os melhores resultados em ambos os casos. No nosso caso, nós apresentado um único método, com quase nenhuma diferença com exceção do conjunto de dados usado para treinamento. Assim sendo, como lição aprendida, acreditamos que em uma competição futura devemos investir mais tempo no ajuste fino do algoritmos para as tarefas e conjuntos específicos.

6 Conclusões e Trabalhos Futuros

Neste artigo apresentamos os métodos e resultados seguidos por nossa equipe na competição ASSIN, e avaliamos os resultados obtidos, em comparação com as outras equipes. No nosso caso, decidimos por explorar abordagens baseadas em vetores de palavra, seguindo duas estratégias distintas: a primeira estratégia é baseada em modelos de regressão tradicionais usando um conjunto de características da literatura para a codificação de similaridade semântica; e a segunda é baseada em redes neurais. Tendo em conta os maus resultados da segunda estratégia nos conjuntos de dados de avaliação, nós prosseguimos na competição somente com o método da primeira estratégia. Com esta abordagem, obtivemos melhores resultados na tarefa de reconhecimento de inferência textual, alcançando o melhor valor de medida F1 no geral, e a melhor acurácia e F1 no conjunto PT-BR. Na tarefa de similaridade semântica, nosso melhor resultado foi o segundo lugar no conjunto PT-BR.

A experiência de participar na competição foi muito valiosa, e esperamos continuar trabalhando nestes problemas para melhorar os nossos métodos e resultados atuais. Dentre os trabalhos futuros, um deles consiste em entender melhor o motivo das redes siamesas não terem apresentado um desempenho tão bom quanto a estratégia baseada nas características de Kenter e Rijke. Além disso, gostaríamos investigar melhor as características de Kenter, a fim de obter melhores resultados nestas tarefas.

Referências

Bentivogli, Luisa, Peter Clark, Ido Dagan, Hoa Trang Dang & Danilo Giampiccolo. 2011. PASCAL recognizing textual entailment challenge (RTE-7) at TAC 2011. Available from http://www.nist.gov/tac/2011/RTE/.

Byun, Hyeran & Seong-Whan Lee. 2002. Applications of support vector machines for pattern recognition: A survey. Em *Proceedings of the First International Workshop on Pattern Recognition with Support Vector Machines*, 213–236.

Chollet, François. 2015. Keras: Theano-based deep learning library. Available from http://keras.io.

Chopra, Sumit, Raia Hadsell & Yann LeCun.

2005. Learning a similarity metric discriminatively, with application to face verification. Em *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 539–546.

Collobert, R., J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu & P. Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12. 2493–2537.

Cortes, Corinna & Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning* 20(3). 273–297.

Hochreiter, Sepp & Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8). 1735–1780.

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

Tibshirani, Robert. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 267–288.

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

FlexSTS: Um Framework para Similaridade Semântica Textual

FlexSTS: A Framework for Semantic Textual Similarity

Jânio Freire
Universidade de Fortaleza
janio.freire@gmail.com

Vládia Pinheiro
Universidade de Fortaleza
vladiacelia@unifor.br

David Feitosa
Universidade de Fortaleza
davidfeitosa@gmail.com

Resumo

Desde 2012, os eventos de *Semantic Evaluation* (SemEval) propõem a tarefa de Similaridade Semântica Textual (STS) como um tema de competição, demonstrando sua relevância. Em 2016, a tarefa foi, pela primeira vez, proposta para língua portuguesa, no Workshop de Avaliação de Similaridade Semântica e Inferência Textual (ASSIN), realizado durante a conferência PROPOR 2016. Neste trabalho, apresentamos o FlexSTS — um *framework* flexível para STS que combina diversos componentes como parsers morfológicos e sintáticos, bases de conhecimento e lexicais, algoritmos de aprendizagem automática, e algoritmos de alinhamento e cálculo da similaridade. Para a ASSIN, FlexSTS foi instanciado em três sistemas de STS para língua portuguesa. Os resultados obtidos foram comparados com uma abordagem *baseline* que utiliza o coeficiente DICE.

Palavras chave

Similaridade Textual, Similaridade Semântica, Avaliação Semântica

Abstract

Since 2012, Semantic Evaluation series (SemEval) propose the task of Semantic Textual Similarity (STS) as a evaluation theme, demonstrating the relevance of this research topic. In 2016, the task was first proposed to the Portuguese language, in the Workshop of Semantic Textual Similarity and Inference Evaluation (ASSIN), held during the conference PROPOR 2016. In this paper, we present the FlexSTS — a flexible framework for STS combining several components as morphological and syntactic parsers, knowledge and lexical databases, machine learning algorithms, and algorithms for alignment and similarity. For ASSIN, FlexSTS was instantiated into three STS systems for Portuguese. The results were compared with a baseline approach that uses DICE coefficient.

Keywords

Textual Similarity, Semantic Similarity, Semantic Evaluation.

1 Introdução

A tarefa de Similaridade Semântica Textual (STS) (Agirre et al., 2013) visa medir o grau de equivalência semântica entre dois textos, capturando a noção de que alguns textos são mais similares que outros. Por exemplo, o par de sentenças "*A organização criminosa é formada por diversos empresários e por um deputado estadual*" e "*Segundo a investigação, diversos empresários e um deputado estadual integram o grupo.*" devem receber um valor de similaridade mais alto que o par de sentenças "*Mas esta é a primeira vez que um chefe da Igreja Católica usa a palavra em público.*" e "*A Alemanha reconheceu ontem pela primeira vez o genocídio armênio*". STS difere das tarefas de Inferência textual (RTE) e Detecção de Paráfrase, principalmente por assumir uma equivalência bidirecional.

Computar a similaridade textual é útil para um número crescente de tarefas de Processamento de Linguagem Natural (PLN) e Inteligência Artificial (IA), tais como a sumarização (Lin & Hovy, 2003) ou o reuso de experiência (Albuquerque et al., 2012).

Desde 2012, os eventos de *Semantic Evaluation* (SemEval)[1] propõem esta tarefa como um tema de competição, demonstrando a relevância da mesma e um tema de pesquisa ainda em aberto. Em 2016, a tarefa foi novamente proposta para língua inglesa na edição do SemEval 2016[2] e, de forma inédita para língua portuguesa, no Workshop de Avaliação de Similaridade Semântica e Inferência Textual (ASSIN), realizado durante a conferência PROPOR 2016[3].

Tradicionalmente, a tarefa consiste em computar o grau de similaridade semântica entre duas sentenças, usando a seguinte escala:

1. Sentenças completamente diferentes, em assuntos diferentes;

[1] https://en.wikipedia.org/wiki/SemEval
[2] http://alt.qcri.org/semeval2016/task1/
[3] http://propor2016.di.fc.ul.pt

2. Sentenças não relacionadas, mas que compactuam do mesmo assunto;
3. Sentenças de certa forma relacionadas, que podem descrever fatos diferentes mas compartilham alguns detalhes;
4. Sentenças fortemente relacionadas, que divergem apenas em alguns detalhes;
5. Sentenças significam exatamente a mesma coisa.

Neste trabalho, apresentamos o FlexSTS — um *framework* genérico que facilita e flexibiliza o desenvolvimento de sistemas de STS, pois combina diversos componentes como *parsers* morfológicos e sintáticos (NLP toolkits), bases de conhecimento e lexicais, algoritmos de aprendizagem automática, e algoritmos de alinhamento e cálculo da similaridade. Especificamente para avaliação no Workshop ASSIN, FlexSTS foi instanciado para língua portuguesa em três configurações (sistemas) usando o parser *Freeling* (Padró & Stanilovsky, 2012), o modelo de similaridade entre palavras HAL (Hyperspace Analog to Language) (Burgess et al., 1998), a base de conhecimento Wordnet (Miller, 1995), o algoritmo de aprendizagem automática proposto por Pedregosa et al. (2011), e o modelo de alinhamento entre termos proposto por Han et al. (2013). Foram enviadas as execuções dos três sistemas de STS e os resultados obtidos foram comparados com uma abordagem *baseline* que utiliza o coeficiente DICE (Rohlf, 1992) de similaridade sintática entre textos. A análise de casos em que nosso melhor sistema não obteve nível de acerto desejado indiciam melhorias para trabalhos futuros.

2 Trabalhos Relacionados

Destacam-se, como estado da arte, os sistemas campeões da tarefa de STS das edições do SemEval 2013, 2014, 2015.

No SemEval 2013, o sistema campeão foi o submetido pela equipe denominada UMBC (Han et al., 2013). Esse sistema consiste de uma abordagem que agrega conhecimento semântico de uma matriz LSA e da WordNet, além de aplicar uma estratégia de alinhamento e penalização, que determina um conjunto de critérios para um mal alinhamento, e valores e a serem descontados para cada tipo de mal alinhamento. O resultado médio da correlação de Pearson foi 0.6181, para língua inglesa.

Em 2014, a equipe vencedora foi a ECNU (Zhao et al., 2014) que utilizou uma abordagem de aprendizagem de máquina com vários algoritmos e 72 *features*. O algoritmo que obteve melhor resultado foi o *Gradient Boosting*. O resultado médio da correlação de Pearson foi 0,8414, também para língua inglesa.

O sistema campeão da edição de 2015 foi apresentado por Sultan et al. (2015) que propôs uma abordagem de aprendizagem de máquina utilizando o algoritmo *Ridge Requession Model*. As características (*features*) definidas para representar o problema baseiam-se na similaridade entre as sentenças, calculada por uma função que usa uma representação vetorial, criada a partir da matriz LSA, de uma base de paráfrase (Ganitkevitch et al., 2013) e da árvore de dependência sintática. Este sistema obteve resultado de 0,8015 (correlação de Pearson).

3 FlexSTS: *Framework* para Similaridade Semântica Textual

Nesta seção apresentamos a proposta do *framework* FlexSTS, o qual define diversos componentes a serem conectados e usados no desenvolvimento de sistemas de STS, agregando modelos e medidas de similaridade, *toolkits* e algoritmos do estado da arte, em cada etapa do processo de STS. A Figura 1 apresenta o fluxo geral do processo de STS e os diversos componentes ou *plugins* necessários.

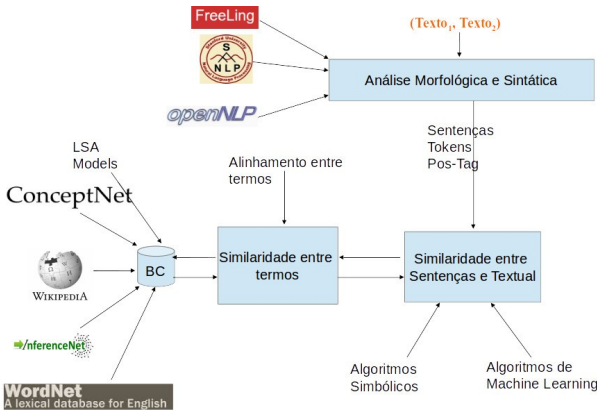

Figura 1: Fluxo do *framework*.

3.1 Análise Morfológica e Sintática

Nesta etapa, dados dois textos de entrada t_1 e t_2, é realizada a detecção das sentenças, a análise morfológica (*tokenização*, lematização, *POS Tagger*) e a análise sintática (*dependency parsing*) de ambos os textos. Inúmeros toolkits disponíveis podem realizar esta tarefa para diversas línguas. Em destaque, tem-se o Stanford NLP Toolkit

(Toutanova et al., 2003), Open NLP (Baldridge, 2005), Freeling (Padró & Stanilovsky, 2012).

O objetivo desta etapa é gerar, para cada texto de entrada, o conjunto de tokens relevantes T_{ij} de cada sentença s_{ij}. O algorıtmo para a construção do conjunto T_{ij}, segue os passos listados abaixo:

1. Análise morfológica e sintática do texto;
2. Reconhecimento de palavras compostas, nomes próprios, valores numéricos, datas e expressões de tempo;
3. Aplicação de heurísticas, seguindo o trabalho de Han et al. (2013):
   (a) Remoção de pontuação;
   (b) Expressões numéricas escritas por extenso são convertidas para números;
   (c) Remoção de *stop words*.
   (d) Referências para tempo são convertidas para o formato militar;
4. Cada *token* das classes abertas de palavras (substantivo, verbo, adverbio e adjetivo), incluindo nomes de entidades reconhecidas, como nomes próprios e abreviações, passam por um processo de desambiguação conforme definido por Pinheiro et al. (2012). Nesse passo, cada termo é associado a um conceito de uma base de conhecimento.
5. Finalmente, o conjunto T_{ij} é formado pelos *tokens* e seus atributos morfológicos, lexicais, sintáticos e semânticos.

3.2 Similaridade Semântica entre Termos

A segunda etapa do processo prevê a aplicação de modelos e medidas para cálculo da similaridade entre palavras $\theta(c, c')$ e de um algoritmo para alinhamento dos termos c e c' de cada sentença s_{1i} e s_{2j} dos textos t_1 e t_2 (textos de entrada).

3.2.1 Modelos de Similaridade Semântica entre Palavras (Word Similarity Models)

O framework define a função $\theta(c, c')$ como uma função parametrizável para vários modelos e medidas de similaridade entre palavras, possibilitando agregar conhecimento adicional expresso em uma ou mais bases de conhecimento e dicionários externos, tais como WikiPedia (Milne & Witten, 2008), WordNet (Miller, 1995), ConceptNet (Liu & Singh, 2004), InferenceNet (Pinheiro et al., 2010a), dentre outras.

Dentre os modelos do estado da arte, tem-se a LSA (*Latent Semantic Analysis*) que segue a hipótese da semântica distribucional, segundo a qual "palavras que ocorrem em contextos similares tendem a ter significados similares" (Harris, 1968). Diversas técnicas de LSA podem ser aplicadas. HAL (*Hyperspace Analog to Language*) (Burgess et al., 1998) é uma técnica de LSA que pode ser aplicada em matriz de co-ocorrência termo-termo. *Singular Value Decomposition* (SVD) (Landauer & Dumais, 1997) tem sido efetiva para melhorar medidas de similaridade entre palavras, visto que podemos selecionar os $k-$maiores valores singulares e reduzir para tamanho k o vetor que representa uma palavra. Por fim, a similaridade entre duas palavras é calculada pela similaridade do cosseno entre os vetores de cada palavra. Han et al. (2013) apresentam uma descrição detalhada do uso do modelo HAL com SVD para língua inglesa.

O modelo de similaridade semântica inferencialista, proposto por Pinheiro et al. (2014) e Pinheiro et al. (2010b) define a *Word Inferential Similarity Measure* a qual calcula a similaridade entre dois conceitos pela interseção entre o conjunto das pré-condições [ou pós-condições] de uso dos dois conceitos, aludindo a ideia de que quanto mais as circunstâncias [ou consequências] de uso de ambos os conceitos são similares, mas as inferências em que os mesmos podem participar são similares.

Han et al. (2013) propõem uma medida de similaridade entre palavras que agrega valor da base WordNet à medida LSA.

3.2.2 Estratégias de Alinhamento entre termos

A estratégia de alinhamento é necessária para definir quais termos de cada sentença serão comparados em termos de similaridade semântica. Considere os textos de entrada t_1 e t_2 com as seguintes sentenças $\{s_{11}, s_{12}, s_{13}\}$ e $\{s_{21}, s_{22}, s_{23}\}$, respectivamente. Na etapa anterior, os conjuntos T_{11} e T_{21} com os termos das sentenças s_{11} e s_{21} foram gerados. Propõe-se então uma função de alinhamento t-align(c) (Fórmula 1 que busca alinhar o termo c em T_{11} com um ou mais termos c' em T_{21}, de acordo com uma das seguintes estratégias:

1. *tokens* de mesma classe gramatical (*POS tag*) (p.ex. substantivo com substantivo, verbo com verbo, etc.);
2. *tokens* com mesma função sintática (p.ex. sujeito com sujeito, verbo principal com verbo principal, etc.);

3. *tokens* com maior valor de similaridade semântica entre palavras;

4. todos os *tokens* com todos;

Seguindo Han et al. (2013), a estratégia 3 alinha o termo c em T_{ij} com o termo c' em T_{lj}, que tiver maior valor de similaridade semântica $\theta(c, c')$ (Fórmula 1).

$$\text{t-align}(c) = \text{argmax}_{c' \in T_{lj}} \theta\left(c, c'\right). \quad (1)$$

A flexibilidade de adotar uma dentre várias estratégias de alinhamento permite adaptar o sistema STS a um domínio ou aplicação. No entanto, argumentamos que a estratégia 1 (que utiliza o critério de *POS tag*) e a estratégia 2 (que utiliza o critério de função sintática) são mais intuitivas e linguisticamente fundamentadas, embora mais complexas.

3.3 Similaridade Semântica Textual

Na última etapa do processo, o framework prevê duas abordagens para cálculo da STS—algoritmos de aprendizagem automática e/ou algoritmos simbólicos.

A abordagem por aprendizagem de máquina preconiza o uso de algoritmos supervisionados, tais como definidos por Chang & Lin (2011), Hall et al. (2009) e Pedregosa et al. (2011), com uso de características (*features*) sintáticas, lexicais e semânticas.

Na abordagem simbólica, a intuição básica de uma medida de similaridade semântica entre textos é que, quanto mais as sentenças dos textos são similares, mais os textos são similares. Da mesma forma, quanto mais os conceitos articulados nas sentenças são similares, mas similares as sentenças também serão. Neste sentido, a medida *SIMt* (Fórmula 4) define a similaridade entre dois textos de entrada t_1 e t_2 pela média da similaridade entre as sentenças s e s' que são mais similares. Ou seja, cada sentença s de t_1, é alinhada com a sentença s' de t_2 que lhe é mais similar.

A Fórmula 2 apresenta nossa função de alinhamento de sentenças s-align(s), a qual, para a sentença s de t_1 (ou t_2), retorna sua contraparte s' em t_2 (ou t_1), com maior valor da medida de similaridade entre sentenças *SIMs* (Fórmula 3).

$$\text{s-align}(s) = \text{argmax}_{s' \in t_i} \text{SIMs}\left(s, s'\right). \quad (2)$$

A Fórmula 3 define a medida de similaridade entre sentenças *SIMs* entre duas sentenças s_1 e s_2 pela média ponderada do somatório das similaridades entre seus termos alinhados.

$$\text{SIMs}\left(s_1, s_2\right) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{q_i} \theta\left(c, c'\right) \times P_i}{\sum_{i=1}^{n} q_i \times P_i} \quad (3)$$

Onde:

- $\theta(c, c')$ é o valor da similaridade entre os *tokens* das sentenças s_1 e s_2, de acordo com o modelo de similaridade entre palavras definido na etapa anterior (seção 3.2.1);
- n é a quantidade de "tipos gramaticais" definidos na estratégia de alinhamento. Por exemplo, usando o critério de alinhamento por função sintática (estratégia 2), pode-se ter $n = 3$, conforme os seguintes tipos: SUJEITO, VERBAL PRINCIPAL e OBJETO;
- q_i é a quantidade de elementos em cada "tipo gramatical" i;
- P_i é o peso do "tipo gramatical" i, permitindo, por exemplo, que a similaridade entre verbos tenha um peso maior que a similaridade entre objetos diretos.

Finalmente, a Fórmula 4 calcula a similaridade semântica entre dois textos de entrada t_1 e t_2, com p e k sentenças, respectivamente.

$$\begin{aligned} \text{SIMt}(t_1, t_2) &= \frac{\sum_{s \in t_1} \text{SIMs}(s, \text{s-align}(s))}{2p} \\ &+ \frac{\sum_{s \in t_2} \text{SIMs}(s, \text{s-align}(s))}{2k} \end{aligned} \quad (4)$$

Pinheiro et al. (2014) apresentam um exemplo ilustrativo de uso das fórmulas acima.

4 Sistemas STS para ASSIN

O framework FlexSTS foi usado para instanciar três sistemas para STS na língua portuguesa, cujos resultados foram submetidos à avaliação no Workshop de Avaliação de Similaridade Semântica e Inferência Textual (ASSIN), realizado durante a conferência PROPOR 2016. A seguir serão explanadas a configuração de cada sistema e do sistema *baseline*. Ao final, os resultados e uma discussão dos mesmos serão apresentados.

Importante aqui salientar a flexibilidade do *framework* FlexSTS onde podem ser mesclados diversos componentes para instanciar ou criar sistemas de STS. Basicamente são selecionados componentes para cada etapa do processo:

Análise Morfológica, Similaridade entre Palavras, e Similaridade entre Textos. As tabelas 1, 2 e 3 apresentadas nas subseções a seguir detalham os componentes utilizados em cada sistema. A escolha dos componentes visou combinar abordagens simbólicas e estatísticas.

4.1 STS_MachineLearning

O sistema STS_MachineLearning aplicou uma abordagem híbrida para cálculo da STS — aprendizagem automática usando dois atributos (*features*) – similaridade entre palavras pelo coeficiente DICE e similaridade entre palavras pela WordNet. A configuração do sistema está descrita na Tabela 1.

Etapa	**Componente / Modelo**	**Ferramenta**
Análise Morfológica/Sintática	POS Tagger / Lematização	FreeLing
Similaridade Semântica de Palavras	Coeficiente DICE	Ver 4.1.1
	WordNet	Ver 4.1.1
Similaridade Semântica Textual	Aprendizagem Automática	Ridge Regression Model

Tabela 1: Configuração do sistema STS_MachineLearning.

4.1.1 Modelo de Aprendizagem de Máquina

No cálculo de STS foi usado o algoritmo *ridge regression model* (Pedregosa et al., 2011), um modelo de regressão com $\alpha = 1.0$ e um resolvedor automático que seleciona o peso de uma coleção dependendo do tipo de dado. Esses algoritmos foram usados por Sultan et al. (2015), campeão da tarefa de STS no SemEval 2015. O treinamento do algoritmo *ridge regression model* foi realizado com o *dataset* de treinamento disponibilizado na ASSIN. A seguir detalhamos os cálculos das duas *features* usadas para caracterizar o conjunto de exemplos.

Feature DICE

Esta *feature* representa a similaridade semântica textual entre os dois textos (exemplo) calculada pela Fórmula 4 usando a coeficiente DICE (Rohlf, 1992) como medida de similaridade entre palavras $\theta(c, c') = \text{DICE}(c, c')$. A Fórmula 5 define este cálculo.

$$\text{DICE}(c, c') = \begin{cases} 1 & \text{se} \begin{cases} isNum(c) \wedge isNum(c') \wedge c = c' \\ isCorrespondingPronoun(c, c') \\ diceCoefficient(c, c') > 2/3 \end{cases} \\ 0 & \text{caso contrário} \end{cases} \quad (5)$$

Onde,

- $isNum(c)$ retorna verdadeiro se c é um número;
- $isCorrespondingPronoun(c, c')$ verifica se os termos c e c' são pronomes correspondentes. Por exemplo, para os pronomes "*eu*" e "*me*" retorna verdadeiro;
- $diceCoefficient(c, c')$ calcula o coeficiente de Dice entre os termos c e c', conforme definido por Rohlf (1992).

Feature WNET

Esta *feature* representa a similaridade semântica textual entre os dois textos (exemplo) calculada pela Fórmula 4 usando conhecimento da WordNet para calcular a similaridade entre palavras, conforme Formula 6:

$$\text{WNET}'(c, c') = 0.5e^{\alpha D(c,c')} \quad (6)$$

Onde,

- $D(c, c')$ é uma função de distância entre os termos na base WordNet, calculado conforme segue:
  - 0, caso os termos pertençam ao mesmo conjunto de sinônimos (*synset*);
  - 1, nos seguintes casos: uma palavra é hiperonímia direta da outra; um adjetivo tem uma relação direta do tipo *similar to* com outro; uma palavra é uma forma derivacional da outra.
  - 2, nos seguintes casos: uma palavra é 2 *links* de hiperonímia indireta da outra; um adjetivo é 2 *links similar to* com outro; uma palavra é cabeça (*head*) do glossário da outra, ou sua hiperônima direta, ou uma das suas hipônimas diretas.
- α, parâmetro de normalização definido por Han et al. (2013) e fixado em 0,25.

A versão utilizada da WordNet foi a versão 3.0 em inglês e foi realizada a tradução dos corpus da

ASSIN (Português-Inglês) pelo Google Tradutor. A escolha desta solução deveu-se a dificuldades técnicas no uso da OpenWordNet.PT[4].

4.2 STS_LSA

O sistema STS_LSA aplicou somente a abordagem simbólica para cálculo da STS, usando o modelo LSA de similaridade entre palavras e a estratégia de alinhamento por termos com maior similaridade (estratégia 3). A configuração do sistema STS_LSA está descrita na Tabela 2

Etapa	**Componente / Modelo**	**Ferramenta**
Análise Morfológica/Sintática	POS Tagger / Lematização	FreeLing
Similaridade Semântica de Palavras	Modelo LSA (HAL+SVD)	Ver 4.2.1
	Estratégia de alinhamento	t-align$_3$ (fórmula 1)
Similaridade Semântica Textual	Altoritmo Matemático STS	Fórmulas 2, 3 e 4

Tabela 2: Configuração do sistema STS_LSA.

4.2.1 Modelo de Similaridade LSA

Foi usada a variação da técnica LSA chamada HAL (*Hyperspace Analog to Language*) (Burgess et al., 1998) que constrói a matriz de coocorrência termo-termo. Para a construção da msubatriz, foi usado o corpus CETENFolha[5] — um corpus de cerca de 24 milhões de palavras em Português-Brasileiro, com base nos textos do jornal Folha de S. Paulo que fazem parte do corpus do Núcleo Interinstitucional de Linguística Computacional (NILC), da USP/São Carlos.

Por questões de desempenho computacional, foram selecionados os 24000 termos que mais ocorrem no corpus, das classes abertas de palavras (substantivos, verbos, adjetivos e advérbios). Neste vocabulário não existem nomes próprios. A frequência de coocorrência entre os 24000 termos foi contada em uma janela de tamanho fixo que passa por todo o corpus. O tamanho de janela utilizado foi ±4, pois foi o que obteve melhor resultado por Han et al. (2013). Por fim, foi aplicada a estratégia de SVD (*Single Value Decomposition*) de Baglama & Reichel (2015), e selecionados os $k = 300$ maiores valores singulares. Assim, o tamanho do vetor que representa as palavras foi reduzido de 24000 para 300. A similaridade entre os termos foi calculada utilizando a função cosseno entre os vetores.

[4] http://wnpt.brlcloud.com/wn/

[5] http://www.linguateca.pt/cetenfolha/

4.3 STS_WORDNET_LSA

O sistema STS_WORDNET_LSA aplicou somente a abordagem simbólica para cálculo da STS, o modelo LSA de similaridade entre palavras e a estratégia de alinhamento por termos com maior similaridade (estratégia 3). Como conhecimento adicional, adicionou informação da WordNet no cálculo da similaridade LSA, a exemplo do trabalho de Han et al. (2013). A configuração do sistema STS_WORDNET_LSA está descrita na Tabela 3.

Etapa	**Componente / Modelo**	**Ferramenta**
Análise Morfológica/Sintática	POS Tagger / Lematização	FreeLing
Similaridade Semântica de Palavras	Modelo LSA (HAL+SVD)	Ver 4.2.1
	Estratégia de alinhamento	t-align$_3$ (fórmula 1)
	Base de Conhecimento / WordNet	Ver 4.3.1
Similaridade Semântica Textual	Altoritmo Matemático STS	Fórmulas 2, 3 e 4

Tabela 3: Configuração do sistema STS_WORDNT_LSA.

4.3.1 LSA + Conhecimento da WordNet

À medida de similaridade entre palavras $\theta(c, c') = \text{LSA}(c, c')$ (ver 3.2.1) foi adicionado conhecimento da base WordNet (Han et al., 2013). A Fórmula 7 apresenta este cálculo.

$$\text{WNET}(c, c') = \text{BASIC}(c, c') + \text{WNET}'(c, c') \quad (7)$$

$$\text{BASIC}(c, c') = \begin{cases} \theta(c, c') & \text{se } \theta \neq nulo \\ \text{DICE}(c, c') & \text{se } usaDice = \top \wedge \\ & (\theta = nulo \vee \theta(c, c') = 0) \\ 0 & \text{caso contário} \end{cases}$$

Onde,

- $\theta(c, c') = \text{LSA}(c, c')$ (ver 3.2.1);
- *usaDice* é um parâmetro que indica se, em caso valor $\theta(c, c')$ nulo ou zerado, deva-se usar o valor do coeficiente DICE;
- $\text{DICE}(c, c')$, conforme definido em Fórmula 5;
- $\text{WNET}'(c, c')$, conforme definido em Fórmula 6.

4.4 STS_Baseline

O sistema *STS_Baseline* foi usado neste trabalho apenas como referência inicial de avaliação, visto que, antes da ASSIN, inexistia estado da arte para STS em língua portuguesa. Nossa proposta foi utilizar o coeficiente de similaridade DICE (conforme definido em 3.1), como sistema *baseline* para a tarefa de STS.

4.5 Resultados e Discussão

A tabela 4 apresenta os resultados da medida de correlação de *Pearson* dos três sistemas STS (*runs*), enviados para ASSIN, após execução no *dataset* de teste para Português-Brasileiro (PT-BR) e Português-Portugal (PT-PT). Nosso melhor sistema foi o STS-MachineLearning em ambos os *datasets*. Na última linha da Tabela 4, apresentamos os resultados do sistema *baseline*, que obteve melhor desempenho que qualquer um dos sistemas avaliados para PT-PT.

Sistema	**PT-BR**	**PT-PT**
STS_MachineLearning	**0,62**	**0,64**
STS_LSA	0,56	0,59
STS_WNET_LSA	0,61	0,63
STS_Baseline	0,60	**0,69**

Tabela 4: Resultados dos sistemas STS desenvolvidos a partir do *framework* FlexSTS.

A seguir elencamos duas dificuldades importantes enfrentadas na construção dos sistemas de STS submetidos à ASSIN:

- No sistema STS_LSA, a matriz de coocorrência termo-termo gerada era muito esparsa, implicando em pouca relevância do cálculo da similaridade pela LSA. Atribui-se como causa o tamanho do corpus e tamanho dos textos do corpus;
- O uso da versão em Inglês da WordNet com a necessidade de solução de tradução Português-Inglês dos corpus ASSIN pode ter prejudicado o desempenho dos sistemas que utilizam esta base.

O uso do sistema *baseline* pelo coeficiente DICE permitiu constatar que uma medida simples de similaridade sintática obteve resultado significativo em relação aos corpus PT-BR (0,60) e PT-PT (0,69). Em apenas 211 casos do corpus *Gold Standard* ASSIN, o valor absoluto da diferença entre o valor da similaridade DICE e o valor GOLD foi superior a 2 ($|DICE - GOLD| > 2$). No demais casos (1935), estes valores são muito próximos. Portanto, conclui-se que os corpus ASSIN possuem uma similaridade lexical alta, dificultando a influência de conhecimento semântico à tarefa de STS.

Analisando alguns casos em que o sistema STS_MachineLearning obteve melhor resultado comparado com a solução baseline (DICE), identificamos que conhecimento semântico agregou valor à tarefa. Por exemplo, para o par de texto t_1 e t_2 na Figura 2, o sistema STS_MachineLearning apresentou valor de similaridade mais correlato ao valor GOLD, pois encontrou valor de similaridade entre as palavras "*intervalo*" e "*tempo*".

> t_1 = "O time treinado por Rafa Benítez assumiu uma postura covarde em o segundo **tempo** e apenas se defendeu"
> t_2 = "O time voltou de o **intervalo** com uma postura covarde e passou a apenas se defender"

Figura 2: Exemplo de textos com uso de conhecimento da WordNet.

5 Conclusão

Neste trabalho apresentamos a proposta do *framework* FlexSTS, o qual define diversos componentes a serem conectados para o desenvolvimento de sistemas de STS, agregando modelos e medidas de similaridade, *toolkits* e algoritmos do estado da arte, em cada etapa do processo de STS.

FlexSTS foi instanciado em três sistemas:

1. STS_MachineLearning: abordagem híbrida para cálculo da STS com aprendizagem automática usando dois atributos (*features*) — similaridade entre palavras pelo coeficiente DICE e similaridade entre palavras pela WordNet;

2. STS_LSA: abordagem simbólica que usa basicamente o modelo de similaridade de palavras da *Latent Semantic Analysis* (LSA);

3. STS_WORDNET_LSA: uma abordagem também simbólica que agrega conhecimento da WordNet à similaridade pela LSA.

Os sistemas foram testados nos *datasets* de teste disponíveis na ASSIN para Português-Brasileiro (PT-BR) e Português-Portugal (PT-PT). Nosso melhor sistema foi o STS-MachineLearning com resultado para o PT-PT de 0,64 (correlação de Pearson). Os principais problemas foram a esparsidade da matriz de coocorrência termo-termo construída a partir do corpus CETEMFolha e o uso da WordNet em inglês. Um resultado importante foi o desempenho do sistema *baseline* pelo coeficiente de DICE, que obteve 0,69 para o *corpus* PT-PT, indiciando que os corpus possuem alta similaridade lexical.

A análise dos resultados, dos problemas enfrentados e de erros do sistema indicam os seguintes trabalhos futuros: criação de mais cenários de testes com diversificação de algoritmos de *machine learning* e novas *features*; construção de nova matriz LSA a partir de um corpus mais robusto na língua portuguesa; agregação de conhecimento da Wikipedia e InferenceNet.

Referências

Agirre, Eneko, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre & Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. Em *Second Joint Conference on Lexical and Computational Semantics (*SEM)*, 32–43.

Albuquerque, Adriano, Vládia Pinheiro & Thiago Leite. 2012. Reuse of experiences applied to requirements engineering: An approach based on natural language processing. Em *Proceedings of the 24th International Conference on Software Engineering & Knowledge Engineering (SEKE'2012)*, 574–577.

Baglama, Jim & Lothar Reichel. 2015. *irlba: Fast truncated svd, pca and symmetric eigen decomposition for large dense and sparse matrices. r package version 2.0.0.*

Baldridge, Jason. 2005. The OpenNLP project. http://opennlp.apache.org.

Burgess, Curt, Kay Livesay & Kevin Lund. 1998. Explorations in context space: Words, sentences, discourse. *Discourse Processes* 25(2–3). 211–257.

Chang, Chih-Chung & Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* 2(3). 27:1–27:27.

Ganitkevitch, Juri, Benjamin Van Durme & Chris Callison-Burch. 2013. PPDB: The paraphrase database. Em *Proceedings of NAACL-HLT*, 758–764.

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

Han, Lushan, Abhay L. Kashyap, Tim Finin, James Mayfield & Johnathan Weese. 2013. UMBC_EBIQUITY-CORE: Semantic Textual Similarity Systems. Em *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*, 44–52. ACL.

Harris, Zellig. 1968. *Mathematical structures of language*. Wiley.

Landauer, Thomas & Susan Dumais. 1997. A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review* 104(2). 211–240.

Lin, Chin-Yew & Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. Em *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1* NAACL '03, 71–78.

Liu, Hugo & Push Singh. 2004. Conceptnet — a practical commonsense reasoning tool-kit. *BT Technology Journal* 22(4). 211–226.

Miller, George A. 1995. Wordnet: A lexical database for english. *Communications of the ACM* 38. 39–41.

Milne, David & Ian H. Witten. 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. Em *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, 25–30.

Padró, Lluís & Evgeny Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. Em *Language Resources Evaluation Conference*, 2473–2479.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake

Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot & Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12. 2825–2830.

Pinheiro, Vládia, Vasco Furtado & Adriano Albuquerque. 2014. Semantic textual similarity of Portuguese-language texts: An approach based on the semantic inferentialism model. Em Jorge Baptista, Nuno Mamede, Sara Candeias, Ivandré Paraboni, Thiago A. S. Pardo & Maria das Graças Volpe Nunes (eds.), *Computational Processing of the Portuguese Language: 11th International Conference*, 183–188.

Pinheiro, Vládia, Vasco Furtado, Lívio Melo Freire & Caio Ferreira. 2012. Knowledge-intensive word disambiguation via commonsense and wikipedia. Em *Proceedings of the 21st Brazilian Conference on Advances in Artificial Intelligence* SBIA'12, 182–191. Springer-Verlag.

Pinheiro, Vladia, Tarcisio Pequeno, Vasco Furtado & Wellington Franco. 2010a. InferenceNet.Br: Expression of inferentialist semantic content of the portuguese language. Em Thiago Alexandre Salgueiro Pardo, António Branco, Aldebaro Klautau, Renata Vieira & Vera Lúcia Strube de Lima (eds.), *Computational Processing of the Portuguese Language: 9th International Conference*, 90–99.

Pinheiro, Vládia, Tarcisio Pequeno & Vasco Furtado. 2010b. Um analisador semântico inferencialista de sentenças em linguagem natural. *Linguamática* 2(1). 111–130.

Rohlf, F. James. 1992. *Numerical taxonomy and multivariate analysis system.* Department of Ecology and Evolution, State University of New York.

Sultan, Md Arafat, Steven Bethard & Tamara Sumner. 2015. Dls@cu: Sentence similarity from word alignment and semantic vector composition. Em *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 148–153.

Toutanova, Kristina, Dan Klein, Christopher D. Manning & Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. Em *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1* NAACL'03, 173–180.

Zhao, Jiang, Tiantian Zhu & Man Lan. 2014. ECNU: one stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. Em *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval-COLING 2014*, 271–277.

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

INESC-ID@ASSIN: Medição de Similaridade Semântica e Reconhecimento de Inferência Textual

INESC-ID@ASSIN: Measuring Semantic Similarity and Recognizing Textual Entailment

Pedro Fialho
Universidade de Évora, INESC-ID
pedro.fialho@l2f.inesc-id.pt

Ricardo Marques
IST/UTL, INESC-ID
ricardo.sa.marques@tecnico.ulisboa.pt

Bruno Martins
IST/UTL, INESC-ID
bruno.g.martins@tecnico.ulisboa.pt

Luísa Coheur
IST/UTL, INESC-ID
luisa.coheur@l2f.inesc-id.pt

Paulo Quaresma
Universidade de Évora, INESC-ID
pq@di.uevora.pt

Resumo

Neste artigo apresentamos o sistema INESC-ID@ASSIN, o qual competiu no evento "Avaliação de Similaridade Semântica e Inferência Textual" (ASSIN) de 2016, nas tarefas de similaridade semântica e reconhecimento de paráfrases (i.e., inferência textual). O sistema INESC-ID@ASSIN aborda o problema de medir a similaridade entre frases como uma tarefa de regressão e aborda a inferência textual como uma tarefa de classificação. Embora o INESC-ID@ASSIN seja baseado essencialmente em características lexicais simples para deteção de paráfrases e reconhecimento de inferência textual, foram obtidos resultados promissores nesta avaliação conjunta.

Palavras chave

aprendizagem supervisionada, regressão, classificação

Abstract

In this article we present INESC-ID@ASSIN, a system that competed in the 2016 joint evaluation effort entitled *Avaliação de Similaridade Semântica e Inferência Textual* (ASSIN), in the tasks of semantic similarity and textual entailment recognition. INESC-ID@ASSIN addresses the problem of detecting sentence similarity as a regression task, and it addresses textual entailment as a classification task. Although INESC-ID@ASSIN relies mainly on simple lexical features for detecting paraphrases and recognizing textual entailment, promising results were achieved in this joint evaluation.

Keywords

supervised learning, regression, classification

1 Introdução

Detetar a quantidade e o tipo de similaridade entre duas frases é uma tarefa complexa de Compreensão de Língua Natural, principalmente devido à variabilidade lexical e sintática característica da língua natural. Detetar equivalência entre frases pode incluir a medição de semelhança semântica, e o problema está também relacionado com as tarefas de identificação de paráfrases ou de inferência textual.

A inferência textual pode ser definida como a tarefa de estimar a relação entre duas unidades de língua natural (por exemplo, entre duas frases), onde a veracidade de uma requer a veracidade da outra. Podemos dizer que de uma frase A se deduz a frase B se e somente se sempre que A é verdade B também é verdade.

Paráfrases são um tipo especial de inferência, nomeadamente inferência bidirecional. Uma paráfrase é uma espécie de equivalência semântica, responsável pela interligação de frases através da substituição de classes gramaticais e mantendo variáveis inalteradas entre as estruturas lexicais e sintáticas.

As tarefas de Identificação de Inferência Textual (RTE, do Inglês Recognizing Textual Entailment) e cálculo da similaridade semântica têm muitas aplicações práticas, sendo usadas em sistemas de pergunta-resposta, para extração de informação, sumarização ou tradução automática (MT, do Inglês Machine Translation), entre outros.

Neste artigo apresentamos o INESC-ID@ASSIN, um sistema que deteta paráfrases e faz inferência textual, baseado em aprendizagem automática supervisionada e que explora propriedades lexicais que relacionam duas frases. Detetar a quantidade de semelhança é conseguido com um modelo de regressão, enquanto o tipo de inferência é previsto com um classificador.

Avaliámos a nossa abordagem no contexto da ASSIN (Avaliação de Similaridade Semântica e Inferência Textual), uma tarefa de avaliação conjunta no PROPOR (Conferência Internacional sobre o Processamento Computacional do Português) de 2016. A tarefa ASSIN forneceu dados de treino e teste com exemplos em Português Europeu (PT-PT) e do Brasil (PT-BR).

O resto deste artigo está organizado da seguinte forma: A Secção 2 apresenta trabalhos relacionados. A Secção 3 apresenta o sistema INESC-ID@ASSIN e a Secção 4 detalha a avaliação e resultados. Finalmente, a Secção 5 conclui e indica trabalho futuro.

2 Trabalho relacionado

O aparecimento de tarefas conjuntas focadas no problema da RTE tem fomentado experiências com várias abordagens baseadas em dados/aprendizagem, aplicadas a tarefas semânticas (Dagan et al., 2009, 2013; Zhao et al., 2014; Bjerva et al., 2014). Particularmente, a disponibilidade de conjuntos de dados para aprendizagem supervisionada tornou possível formular o problema da RTE como uma tarefa de classificação, em que características são extraídas a partir dos exemplos de treino e utilizadas pelos algoritmos de aprendizagem automática na construção de um classificador, que é finalmente aplicado aos dados de teste.

Abordagens recentes para RTE ou para a identificação de paráfrases utilizam algoritmos de aprendizagem automática (por exemplo, classificadores lineares) com uma variedade de características, baseadas em comparações sobre padrões lexicais, sintáticos e/ou semânticos, contagem de co-ocorrências em documentos, e regras de primeira ordem para reescrita sintática.

Diferentes abordagens têm sido formuladas, muitas vezes envolvendo a combinação de características como as acima descritas. Uma abordagem simples é a estratégia saco-de-palavras, em que a semelhança de um par de frases é calculada utilizando a similaridade do cosseno entre representações vetoriais. Se o valor da similaridade é superior a um limiar pré definido (estabelecido manualmente ou aprendido através de dados) as frases são classificadas como paráfrases.

Zhang & Patrick (2005) propuseram um método de classificação em que o par de frases é simplificado para formas canónicas através de regras para alterar a voz passiva para ativa, entre outras. Utilizando árvores de decisão, os autores exploram características baseadas em comparações lexicais, tais como a distância de edição entre símbolos (e.g., letras ou palavras).

Além de utilizar características de comparação lexical, autores como Kozareva & Montoyo (2006) ou Ul-Qayyum & Wasif (2012) propuseram abordagens baseadas em classificação utilizando uma combinação de características lexicais, semânticas e heurísticas (por exemplo, padrões de negação) para auxiliar a deteção de falsas paráfrases.

Os métodos utilizados na maioria das anteriores abordagens funcionam ao nível das frases, mas visto que as paráfrases utilizam tipicamente sinónimos ou outras formas de palavras relacionadas, autores como Mihalcea et al. (2006) ou Fernando & Stevenson (2008) desenvolveram métodos de similaridade ao nível de palavras para determinar se uma frase é paráfrase de outra. Estes métodos são baseados em medidas de similaridade palavra-a-palavra (por exemplo, métricas baseadas em dados que utilizem a WordNet). Métodos baseados em alinhamentos (como os formulados para sumarização ou tradução) são também usuais.

Madnani et al. (2012) propuseram uma abordagem baseada em métricas para alinhamento de sequências de caracteres, utilizadas em tradução automática (MT). Embora o uso de métricas de MT para a tarefa de identificação de paráfrases não seja novidade (Finch et al., 2005), o mérito dos autores está na re-avaliação dessas métricas, conjuntamente com a criação de novas métricas, alcançando um dos melhores resultados sobre o conhecido Microsoft Research Paraphrase Corpus (Dolan et al., 2004).

Pakray et al. (2011) descrevem uma abordagem lexical e sintática para resolver o problema da RTE. Este método resulta da composição de vários módulos, nomeadamente módulos de pré-processamento, similaridade lexical e similaridade sintática.

Tsuchida & Ishikawa (2011) propuseram um sistema RTE que usa métodos de aprendizagem automática com características baseadas em informação lexical e ao nível das estruturas predicado-argumento. A ideia subjacente é delimitar os pares texto-hipótese identificados como tendo inferência textual, mas que na verdade não

têm, ou seja, falsos positivos classificados pelo módulo de nível lexical podem ser rejeitados pelo módulo de nível da frase.

É importante notar que os trabalhos anteriores normalmente correspondem a métodos que são independentes do idioma pelo uso de estratégias simples, tal como a contagem *n*-gramas. Da maioria das abordagens RTE descritas também se conclui que os módulos lexicais alcançam melhores resultados do que os módulos sintáticos e baseados na estrutura de frases.

As mais recentes abordagens a estes problemas dependem de recursos dependentes do idioma e, como seria de esperar, focam-se na língua Inglesa, explorando modelos de semântica distribuída, utilizando recursos como word embeddings (Cheng & Kartsaklis, 2015). Apenas muito recentemente foram publicados recursos que permitiriam replicar algumas destas experiências tendo em conta o Português (por exemplo, (Rodrigues et al., 2016)).

3 INESC-ID@ASSIN

Os modelos de regressão/classificação gerados no contexto do INESC-ID@ASSIN foram baseados no formalismo dos *kernel methods* e usam várias métricas de similaridade. Vários estudos anteriores, na área de Processamento de Língua Natural (NLP, do Inglês Natural Language Processing) e também em outros domínios, usaram métodos semelhantes para combinar múltiplas métricas de similaridade no contexto de obter a semelhança entre objetos (Martins, 2011; Madnani et al., 2012).

As métricas usadas para extrair características dos dados têm em conta, em especial, contribuições da informação lexical. Algumas destas métricas inspiram-se em estudos focados na identificação de paráfrases; outras em estudos relativos a RTE. Várias formas de representação do texto são tidas em conta (minúsculas, Metaphone, etc.).

Os recursos utilizados no INESC-ID@ASSIN são explicados nas seguintes secções e descritos mais detalhadamente em Marques (2015). Uma máquina de suporte de vectores (do Inglês *Support Vector Machine* (SVM)) foi utilizada para a classificação (RTE e identificação de paráfrases) e um modelo do tipo *Kernel Ridge Regression* (KRR) foi utilizado para obter valores contínuos (quantificação de similaridade). Usamos as implementações SVM/KRR do pacote de ferramentas scikit-learn[1], para Python.

[1] http://scikit-learn.org/

3.1 Similaridade lexical

As características de comparação lexical consideradas no INESC-ID@ASSIN são as seguintes:

1. **Maior Subsequência Comum**. O tamanho da maior subsequência comum (LCS) entre o texto e a hipótese. O valor é fixado entre 0 e 1, dividindo o tamanho da LCS pelo tamanho da frase mais longa.
2. **Distância de edição**. A distância mínima de edição/alteração entre símbolos (letras ou palavras) do texto e da hipótese.
3. **Comprimento**. A diferença absoluta de comprimento (número de símbolos) entre o texto e a hipótese. Os comprimentos máximo e mínimo são também considerados (separadamente) como características.
4. **Similaridade por Cosseno**. A similaridade do cosseno entre o texto e a hipótese, com base no número de ocorrências de cada palavra no texto/hipótese (a representação usa a frequência dos termos nos vetores associados a cada documento). A fórmula do cosseno é mostrada na Equação 1.

$$\cos(s_1, s_2) = \frac{\overrightarrow{V}(s_1) \cdot \overrightarrow{V}(s_2)}{||\overrightarrow{V}(s_1)|| \times ||\overrightarrow{V}(s_2)||} \quad (1)$$

   O resultado é um número contínuo entre 0 e 1. Quanto maior o valor, maior a semelhança no par texto-hipótese.
5. **Similaridade de Jaccard**. A similaridade de Jaccard entre o texto e a hipótese. O valor retornado é um número contínuo entre 0 e 1, onde 1 significa que as frases são iguais, e 0 que são totalmente diferentes. O coeficiente de similaridade de Jaccard é usado para comparar a semelhança e diversidade de conjuntos. Mede a semelhança entre conjuntos finitos, e é definido como a divisão entre o número de elementos na intersecção e na união dos conjuntos. A similaridade de Jaccard entre dois conjuntos de palavras s_1 e s_2 é assim definida da seguinte forma:

$$\text{Jaccard}(s_1, s_2) = \frac{|s_1 \cap s_2|}{|s_1 \cup s_2|} \quad (2)$$

6. **Soft TF-IDF**. A métrica Soft TF-IDF mede a similaridade entre representações vetoriais das frases, mas considerando uma métrica de similaridade interna para encontrar palavras equivalentes. A métrica Jaro-Winkler para

similaridade entre palavras, com um limiar de 0.9, é utilizada como métrica de similaridade interna. A distância $\text{Jaro}(s_1, s_2)$ entre duas sequências s_1 e s_2 é:

$$\text{Jaro}(s_1, s_2) = \begin{cases} 0 & \text{se } m = 0 \\ \frac{1}{3} \times \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) & \text{se } m \neq 0 \end{cases} \quad (3)$$

Na equação, m é o número de caracteres coincidentes e t é metade do número de transposições. A métrica Jaro-Winkler modifica a métrica Jaro adicionando-lhe mais peso quando há um prefixo em comum. Este melhoramento define 2 variáveis: (1) *PL*, o comprimento do maior prefixo comum entre duas sequências, com um limite de 4, e (2) *PW*, o peso a dar ao prefixo.

$$\text{JaroWinkler}(s_1, s_2) = (1 - \text{PL} \times \text{PW}) \times \text{Jaro}(s_1, s_2) + \text{PL} \times \text{PW} \quad (4)$$

3.2 Características sobre RTE

As características inspiradas em estudos com o foco em RTE são as seguintes:

1. **Sobreposição NE**. A similaridade de Jaccard considerando apenas entidades mencionadas (NE – do Inglês Named Entities). Para simplificar, entidades mencionadas são todas as palavras que contêm letras maiúsculas.
2. **Sobreposição NEG**. A similaridade de Jaccard considerando apenas palavras negativas. As palavras negativas são: *não*, *nunca*, *jamais*, *nada*, *nenhum*, *ninguém*.
3. **Sobreposição Modal**. A similaridade de Jaccard considerando apenas palavras modais. As palavras modais são: *podia*, *poderia*, *dever*, *deve*, *devia*, *deverá*, *deveria*, *faria*, *possível*, *possibilidade*, *possa*.

3.3 Características sobre paráfrases

As características inspiradas em estudos sobre identificação de paráfrases são as seguintes:

1. **BLEU**. Esta métrica de MT corresponde à quantidade de sobreposições em n-gramas, para diferentes valores de n, entre duas frases, ajustada por uma penalização relativa ao seu comprimento (Papineni et al., 2002). O maior n que utilizámos foi 3, para a cobertura de frases curtas, visto que é sugerido em Papineni et al. (2002) que este valor produz um desempenho semelhante, em comparação com o valor clássico de 4-gramas (BLEU-4).
2. **METEOR**. Esta métrica é uma variação do BLEU com base na média harmónica da precisão e cobertura de unigramas, tendo a cobertura maior peso do que a precisão (Banerjee & Lavie, 2007).
3. **TER**. A Taxa de Erros de Tradução (TER) é uma extensão da Taxa de Erros em Palavras (ou Word Error Rate — WER), que é uma métrica simples baseada em programação dinâmica e que é definida como o número de alterações necessárias para transformar uma sequência noutra. A TER inclui um algoritmo heurístico para lidar com transposições, além de inserções, remoções e substituições (Snover et al., 2006).
4. **NCD**. A Distância de Compressão Normalizada (NCD) é uma forma geral de medir a similaridade entre dois objetos (Li et al., 2004). A ideia subjacente é que ao compactar duas sequências s_1 e s_2 somente a informação sobreposta é extraída.
5. **ROUGE-N**. Sobreposição de n-gramas com base em estatísticas de co-ocorrência (Lin & Hovy, 2003).
6. **ROUGE-L**. Uma variação da métrica ROUGE-N com base no comprimento da maior subsequência de palavras comum (Lin & Och, 2004).
7. **ROUGE-S**. Uma variação da métrica ROUGE-N baseada em skip-bigrams (ou seja, bigramas/pares de palavras, pela ordem em que ocorrem na frase, e possibilitando intervalos entre as palavras) (Lin & Och, 2004).

3.4 Características numéricas

A inspiração para estas características numéricas é simples: frases que se referem às mesmas entidades, mas com números diferentes, são suscetíveis de ser contraditórias. O cálculo desta característica é simples, resultando da multiplicação de 2 similaridades de Jaccard. Uma entre os caracteres numéricos no par texto-hipótese, e outra entre as palavras em torno de tais caracteres numéricos. O resultado é um valor contínuo entre 0 e 1, onde 0 indica que as frases são possivelmente contraditórias.

3.5 Representações de texto

As características anteriormente descritas são aplicadas a diferentes representações das frases. Nomeadamente, considerámos as seguintes representações:

1. **Símbolos originais**.
2. **Símbolos em minúsculas**.
3. **Símbolos em minúsculas sem variações terminais (obtidos pela aplicação de um algoritmo de *stemming*)**.
4. **Agrupamentos de palavras**. O algoritmo de Brown para o agrupamento de palavras é um método aglomerativo que agrega palavras numa árvore binária de classes (Turian et al., 2010), através de um critério baseado na probabilidade logarítmica de um texto perante um modelo de língua baseado em classes.

   O procedimento de agrupamento de Brown foi aplicado a uma coleção de documentos noticiosos do jornal Português *Público*, do qual resultaram 1001 agrupamentos. Nesta representação, as palavras/símbolos são substituídos pelas classes correspondentes.
5. **Double Metaphone**. Foi utilizado um algoritmo bem conhecido para codificar palavras com base na sua fonética, interpretando cada palavra como uma combinação dos sons de 12 consoantes. No entanto, importa referir que o algoritmo Double Metaphone (Philips, 1990) é baseado na pronúncia Inglesa, sendo mais adequado para codificar palavras em inglês e palavras estrangeiras tipicamente utilizadas nos Estados Unidos.
6. **Trigramas de caracteres**. Os trigramas são um caso especial do conceito de n-grama, onde n é 3. Os trigramas de caracteres são usados como termos-chave numa representação da frase, à semelhança de como as palavras são usadas como termos-chave para representar um documento.

Os nossos modelos combinam características com base nestas diferentes representações, considerando um total de 96 características. Algumas características não são adequados para serem combinadas com algumas representações, tal como a característica numérica com a representação Double Metaphone. As combinações consideradas são descritas na Tabela 1.

Feature	O	L	S	C	DM	T
LCS	✓	✓	✓	✓	✓	
D. de edição	✓	✓	✓	✓	✓	
Cosseno	✓	✓	✓	✓	✓	✓
C. Absoluto	✓	✓	✓	✓	✓	
C. Máximo	✓	✓	✓	✓	✓	
C. Mínimo	✓	✓	✓	✓	✓	
Jaccard	✓	✓	✓	✓	✓	✓
Soft TF-IDF	✓	✓	✓			
NE	✓	✓	✓	✓	✓	✓
NEG	✓	✓	✓	✓	✓	✓
Modal	✓	✓	✓	✓	✓	✓
BLEU-3	✓	✓	✓	✓	✓	
METEOR	✓	✓	✓	✓	✓	
ROUGE N	✓	✓	✓	✓	✓	
ROUGE L	✓	✓	✓	✓	✓	
ROUGE S	✓	✓	✓	✓	✓	
TER	✓	✓	✓	✓	✓	
NCD	✓	✓	✓	✓	✓	
Numérica	✓	✓	✓			

Tabela 1: Combinação de características com representações, onde O, L, S, C, DM e T correspondem a símbolos originais, minúsculas, sem terminações, agrupamentos, Double Metaphone e trigramas, respetivamente.

4 Avaliação

O INESC-ID@ASSIN foi avaliado no conjunto de dados ASSIN para medir o seu desempenho na tarefa de quantificar automaticamente a similaridade semântica e tipo de inferência textual.

Reportamos resultados de 2 configurações distintas, uma utilizando um kernel polinomial em modelos SVM e KRR e outra utilizando um kernel linear. Para os modelos lineares, as características mais informativas também são reportadas.

Cada experiência gerou resultados para 3 configurações diferentes, em ambas as tarefas e para dados de teste portugueses e brasileiros.

Além disso, também medimos o desempenho ao treinar o nosso sistema com uma variedade do Português e testar com a outra.

As configurações diferem nos dados utilizados para treino dos algoritmos de aprendizagem. Um desses conjuntos de dados corresponde à expansão do ASSIN com frases traduzidas automaticamente desde o corpus SICK (Marelli et al., 2014), enquanto que as restantes configurações usam partições do ASSIN original.

4.1 Descrição da Tarefa

O ASSIN contém 10000 pares de frases recolhidas de Google News, particionados em conjuntos de treino e teste, com um número de exemplos portugueses e brasileiros igualmente distribuído por cada conjunto. Cada par de frases é anotado para similaridade semântica e inferência textual.

A similaridade semântica é um valor contínuo de 1 a 5, de acordo com as seguintes diretrizes sobre as frases de um par:

1. Completamente diferentes, sobre diferentes assuntos;
2. Não relacionadas, mas mais ou menos sobre o mesmo assunto;
3. Algo relacionadas. Podem descrever factos diferentes, mas partilham alguns detalhes;
4. Fortemente relacionadas, mas alguns detalhes são diferentes;
5. Essencialmente a mesma coisa.

A anotação da inferência textual é uma atribuição categórica usando classes que identificam inferência, paráfrase ou nenhuma relação.

O ASSIN define 2 tarefas para quantificar/calcular a similaridade semântica e classificar o tipo de inferência textual. O desempenho é medido separadamente para as variantes de Portugal e do Brasil.

4.2 Treinar com mais dados

Experimentámos utilizar métodos de MT para expandir o conjunto de dados ASSIN original com novas frases de um conjunto de dados em Inglês, visto que mais dados normalmente conduzem a melhores resultados.

O conjunto de dados SICK (Marelli et al., 2014) é muito semelhante ao ASSIN, em tamanho e tipo de anotações. No entanto, é baseado em legendas de imagens e vídeos, obtidas por crowdsourcing, logo representa menor variabilidade linguística mas mais similaridade entre pares (ou seja, mais pares similares).

O SICK foi traduzido para Português, usando um programa Python assente no serviço de tradução online Microsoft Bing, e conjugado com os conjuntos de treino em português europeu e brasileiro. Assim, adicionamos 9191 exemplos do SICK aos 6000 exemplos do ASSIN, para uma das configurações.

4.3 Resultados

A nossa abordagem à tarefa ASSIN foi avaliada utilizando o coeficiente de Pearson e o erro quadrático médio (MSE) como métricas para similaridade semântica, e com a Exatidão e a medida F1 para RTE.

Consideramos 3 configurações/tentativas diferentes para a nossa abordagem, que diferem na quantidade de dados de treino que são usados, nomeadamente:

1. PT-PT or PT-BR: treinar apenas com dados da mesma variedade de Português (Europeu ou do Brasil, respetivamente) dos dados de teste (3000 exemplos).
2. AllPT: juntar os dados de ambas as variedades para treino, independentemente do teste pretendido (6000 exemplos).
3. PT+BingSICK: usar ambas as variedades e os dados do SICK traduzido para treino (15191 exemplos, dos quais 9191 são do SICK).

Estas configurações foram avaliadas nos dados de teste europeus e brasileiros, embora na entrega oficial só tenha sido avaliado o teste europeu. Na entrega oficial, PT com um kernel polinomial foi a nossa melhor configuração (nos dados de teste europeus). No entanto, devido a um problema no software (agora resolvido) os valores oficiais foram inferiores aos apresentados na Tabela 2.

Os resultados para a nossa abordagem à tarefa ASSIN, recorrendo a um kernel polinomial, são apresentados nas Tabelas 2 e 3.

	Similaridade		RTE	
Treino	**Pearson**	**MSE**	**Exatidão**	**F1**
PT-PT	0.74	0.60	83.55%	0.68
AllPT	0.74	0.60	83.95%	0.69
PT+BingSICK	0.72	0.68	80.70%	0.59

Tabela 2: Resultados da avaliação, com um kernel polinomial e considerando todas as características — teste europeu.

	Similaridade		RTE	
Treino	**Pearson**	**MSE**	**Exatidão**	**F1**
PT-BR	0.73	0.36	85.45%	0.64
AllPT	0.73	0.36	85.70%	0.66
PT+BingSICK	0.70	0.40	84.30%	0.58

Tabela 3: Resultados da avaliação, com um kernel polinomial e considerando todas as características — teste brasileiro.

Os resultados para a nossa abordagem à tarefa ASSIN, recorrendo a um kernel linear, são apresentados nas Tabelas 4 and 5.

	Similaridade		RTE	
Treino	**Pearson**	**MSE**	**Exatidão**	**F1**
PT-PT	0.73	0.62	84.90%	0.71
AllPT	0.74	0.61	84.05%	0.68
PT+BingSICK	0.70	0.73	77.10%	0.47

Tabela 4: Resultados da avaliação, com um kernel linear e considerando todas as características — teste europeu.

	Similaridade		RTE	
Treino	**Pearson**	**MSE**	**Exatidão**	**F1**
PT-BR	0.73	0.36	85.35%	0.55
PT	0.73	0.36	85.85%	0.66
PT+BingSICK	0.70	0.42	82.60%	0.46

Tabela 5: Resultados da avaliação, com um kernel linear e considerando todas as características — teste brasileiro.

O desempenho com um kernel linear é semelhante ao de um kernel polinomial, mas a vantagem da maior dimensionalidade do espaço de um kernel polinomial é realçada quando existem mais dados, como pode ser visto na queda de desempenho dos modelos lineares quando se utiliza o conjunto de dados expandido com MT (em particular no MSE e F1), comparando com os resultados obtidos com um kernel polinomial.

Destes resultados podemos concluir que utilizar dados de treino selecionados/verificados (manualmente) pode melhorar ligeiramente o desempenho, enquanto que dados de treino não filtrados (repetitivos e com erros lexicais ou sintáticos resultantes de MT) prejudica o desempenho da nossa abordagem.

Comparando os resultados por tabela, a configuração que mais consistentemente tem os melhores resultados é a AllPT, tanto para RTE como para medição da similaridade. Considerando todas as tabelas, o nosso sistema tem melhor desempenho nos dados da variante do Brasil.

Os restantes sistemas que participaram na tarefa ASSIN obtiveram resultados inferiores aos apresentados. Barbosa et al. (2016) experimenta SVM e redes neuronais em características baseadas em word embeddings, e apresenta uma visão geral dos resultados obtidos por todos os sistemas que participaram no ASSIN.

Em (Hartmann, 2016) são utilizadas características baseadas em conjuntos de palavras (logo esparsas), onde também figuram os word embeddings. Este sistema obteve os resultados mais próximos dos descritos neste artigo, embora só tenha participado na medição de similaridade semântica.

A abordagem de Freire et al. (2016) introduz um conjunto de ferramentas para sistemas de similaridade entre frases, instanciado com semântica distribuída e conhecimento da WordNet. Este sistema também não participou na medição de similaridade semântica.

Por último, o sistema de Alves et al. (2016) apresenta uma abordagem não supervisionada, individualmente e como característica de uma abordagem supervisionada. Os piores resultados são da abordagem não supervisionada, enquanto que a supervisionada atingiu resultados semelhantes aos de Barbosa et al. (2016), e os mais próximos dos resultados reportados neste artigo relativamente a RTE.

Experimentámos também compreender o desempenho dos modelos treinados com uma variedade de Português e testados com a outra variedade. Como apresentado na Tabela 6, compreender uma variedade do Português conhecendo apenas a outra é melhor do que utilizando o conjunto de dados SICK, traduzido automaticamente pelo sistema Bing. Para simplificar, só é apresentada a experiência com kernels polinomiais, mas com kernels lineares foram obtidos resultados semelhantes.

	Similaridade		RTE	
Treino	**Pearson**	**MSE**	**Exatidão**	**F1**
PT-BR	0.73	0.63	82.70%	0.64
PT-PT	0.72	0.37	84.30%	0.66

Tabela 6: Variando o conjunto de treino e testando com a outra/restante variedade do Português, utilizando um kernel polinomial e todas as características.

4.4 Melhores características

Utilizamos o método Recursive Feature Elimination, tal como implementado no scikit-learn, para obter as 10 melhores características com a configuração PT (i.e., a que produziu os melhores resultados), para cada tarefa (RTE e quantificação de similaridade).

Este é um método para seleção de características com base no seu peso relativamente ao modelo. Como o scikit-learn só representa os pesos das característica em modelos com kernels lineares, apenas aplicamos seleção de características nos nossos modelos lineares.

As 10 melhores características para RTE (classificação) são:

- Soft TF-IDF, em símbolos originais;
- Jaccard, sobre Double Metaphone;
- Jaccard, sobre símbolos em minúsculas sem variações terminais;
- Comprimento Absoluto, em Double Metaphone;
- LCS, sobre símbolos em minúsculas sem variações terminais;
- Numérica, em símbolos originais;
- Sobreposição NE, em Double Metaphone;
- ROUGE-N, em símbolos originais;
- ROUGE-L, sobre símbolos em minúsculas sem variações terminais;
- TER, sobre símbolos em minúsculas sem variações terminais.

As 10 melhores características para quantificação de similaridade (regressão) são:

- Similaridade do Cosseno, em símbolos originais;
- Soft TF-IDF, em símbolos originais;
- Jaccard, em Double Metaphone;
- Jaccard, sobre símbolos em minúsculas sem variações terminais;
- Jaccard, em trigramas de caracteres;
- Numérica, sobre símbolos em minúsculas sem variações terminais;
- Sobreposição NE, em Double Metaphone;
- ROUGE-N, sobre símbolos originais;
- ROUGE-N, em agrupamentos de palavras;
- ROUGE-S, sobre símbolos em minúsculas sem variações terminais.

As características baseadas em similaridade lexical contribuem para os melhores resultados de ambas as tarefas, em especial se se tiver em conta as representações que mantêm os símbolos da frase, como comprovado pela predominância destas métricas e representações entre as 10 melhores características. A única característica baseada em RTE que teve um desempenho relevante é a Sobreposição NE, sobre a representação de texto processado pelo algoritmo Double Metaphone.

5 Conclusões e trabalho futuro

Este trabalho tem por foco as tarefas de RTE e de quantificação de similaridade textual, abordando as mesmas através da aplicação de várias características baseadas em trabalhos anteriores para RTE e identificação de paráfrases - essencialmente métricas provenientes dos domínios de MT e sumarização. Estas características, juntamente com outras relativas a similaridade entre sequências e aspetos numéricos, representam uma nova abordagem que se afasta da mais recente tendência da área, que essencialmente se foca em sistemas baseados em alinhamentos semânticos e correspondência entre relações binárias.

Como trabalho futuro, iremos começar por comparar o desempenho do sistema INESC-ID@ASSIN com variantes, usando os mesmos algoritmos de aprendizagem, aplicados a características mais complexas baseadas em representações sintáticas/semânticas e baseadas em fontes de conhecimento enriquecidas.

Agradecimentos

Este trabalho foi suportado por fundos nacionais através da Fundação para a Ciência e a Tecnologia (FCT), através do projeto com referência UID/CEC/50021/2013. O trabalho foi ainda suportado pelo projeto internacional RAGE com referência H2020-ICT-2014-1/644187 e pelo projeto LAW-TRAIN com referência H2020-EU.3.7.-653587.

Referências

Alves, Ana Oliveira, Ricardo Rodrigues & Hugo Gonçalo Oliveira. 2016. ASAPP: alinhamento semântico automático de palavras aplicado ao português. *Linguamática* 8(2). 43–58.

Banerjee, Satanjeev & Alon Lavie. 2007. METEOR: An automatic metric for mt evaluation with improved correlation with human judgments. Em *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 228–231.

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

Bjerva, Johannes, Johan Bos, Rob van der Goot & Malvina Nissim. 2014. The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity.

Em *Proceedings of the International Workshop on Semantic Evaluation*, 642–646.

Cheng, Jianpeng & Dimitri Kartsaklis. 2015. Syntax-aware multi-sense word embeddings for deep compositional models of meaning. Em *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1531–1542.

Dagan, Ido, Bill Dolan, Bernardo Magnini & Dan Roth. 2009. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering* 15(04). i–xvii.

Dagan, Ido, Dan Roth, Mark Sammons & Fabio Massimo Zanzotto. 2013. Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies* 6(4). 1–220.

Dolan, Bill, Chris Quirk & Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. Em *Proceedings of the International Conference on Computational Linguistics*, s. pp.

Fernando, Samuel & Mark Stevenson. 2008. A semantic similarity approach to paraphrase detection. Em *Proceedings of the Annual Research Colloquium on Computational Linguistics in the UK*, s. pp.

Finch, Andrew, Young-Sook Hwang & Eiichiro Sumita. 2005. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. Em *Proceedings of the International Workshop on Paraphrasing*, 17–24.

Freire, Jânio, Vládia Pinheiro & David Feitosa. 2016. FlexSTS: Um framework para similaridade semântica textual. *Linguamática* 8(2). 23–31.

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

Kozareva, Zornitsa & Andres Montoyo. 2006. Paraphrase identification on the basis of supervised machine learning techniques. Em *Proceedings of the International Conference on Advances in Natural Language Processing*, 524–533.

Li, Ming, Xin Chen, Xin Li, Bin Ma & Paul Vitányi. 2004. The similarity metric. *Information Theory, IEEE Transactions on* 50(12).

Lin, Chin-Yew & Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. Em *Proceedings of the Conference of the North American Chapter of the ACL on Human Language Technology*, 71–78.

Lin, Chin-Yew & Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. Em *Proceedings of the Annual Meeting of ACL*, s. pp.

Madnani, Nitin, Joel Tetreault & Martin Chodorow. 2012. Re-examining machine translation metrics for paraphrase identification. Em *Proceedings of the Conference of the North American Chapter of ACL*, 182–190.

Marelli, Marco, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi & Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. Em *Proceedings of the International Conference on Language Resources and Evaluation*, 216–223.

Marques, Ricardo. 2015. *Detecting contradictions in news quotations*: IST, University of Lisbon. Tese de Mestrado.

Martins, Bruno. 2011. A supervised machine learning approach for duplicate detection over gazetteer records. Em *Proceedings of the International Conference on GeoSpatial Semantics*, 34–51.

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

Pakray, Partha, Sivaji Bandyopadhyay & Alexander Gelbukh. 2011. Textual entailment using lexical and syntatic similarity. *Internacional Journal of Artificial Intelligence and Applications* 2(1). 43–58.

Papineni, Kishore, Salim Roukos, Todd Ward & Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. Em *Proceedings of the Annual Meeting of ACL*, 311–318.

Philips, L. 1990. Hanging on the metaphone. *Computer Language Magazine* 7(12). 39–44.

Rodrigues, João António, António Branco, Steven Neale & João Ricardo Silva. 2016. Lx-dsemvectors: Distributional semantics models for portuguese. Em *Computational Processing of the Portuguese Language - 12th International Conference, PROPOR 2016*, 259–270.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla & John Makhoul. 2006. A study of translation edit rate with targeted human annotation. Em *Proceedings of the Conference of the Association for Machine Translation in the Americas*, 223–231.

Tsuchida, Masaaki & Kai Ishikawa. 2011. A method for recognizing textual entailment using lexical-level and sentence structure-level features. Em *Proceedings of the Text Analysis Conference*, s. pp.

Turian, Joseph, Lev Ratinov & Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. Em *Proceedings of the Annual Meeting of ACL*, 384–394.

Ul-Qayyum, Zia & Altaf Wasif. 2012. Paraphrase identification using semantic heuristic features. *Research Journal of Applied Sciences, Engineering and Tecnhology* 4(22). 4894–4904.

Zhang, Yitao & Jon Patrick. 2005. Paraphrase identification by text canonicalization. Em *Proceedings of the Australasian Language Technology Workshop*, 160–166.

Zhao, Jiang, Tiantian Zhu & Man Lan. 2014. ECNU: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. Em *Proceedings of the International Workshop on Semantic Evaluation*, 271–277.

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

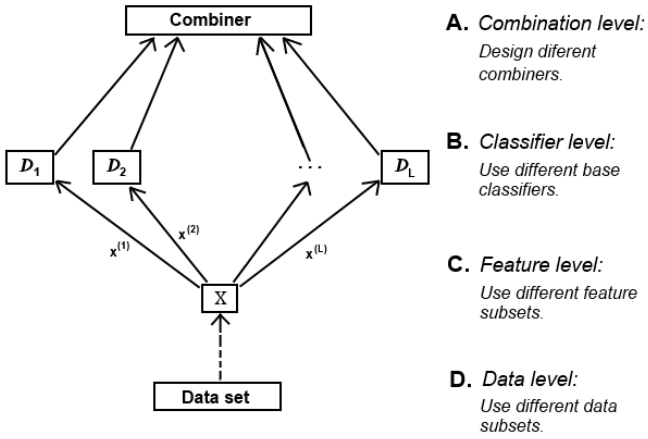

Figura 1: Abordagens para criar conjuntos de classificadores/modelos de regressão (em inglês *ensembles*) (Kuncheva, 2004)

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.

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

Solo Queue at ASSIN: Combinando Abordagens Tradicionais e Emergentes

Solo Queue at ASSIN: Mix of Traditional and Emerging Approaches

Nathan Siegle Hartmann
Universidade de São Paulo
nathansh@icmc.usp.br

Resumo

No presente artigo apresentamos uma proposta para atribuição automática da similaridade entre duas sentenças, tarefa definida na avaliação conjunta ASSIN 2016. Nossa proposta consiste no uso de uma *feature* clássica da classe *bag-of-words*, a TF-IDF; e uma *feature* emergente, capturada por meio de *word embeddings*. Sabe-se que a medida TF-IDF pode ser utilizada para relacionar documentos que contém os mesmos elementos e, portanto, pode ser utilizada para documentos que compartilham palavras. *Word embeddings* é uma técnica de semântica distribucional e é conhecida por modelar a sintaxe e semântica das palavras e, segundo Mikolov et al. (2013a), pode ser utilizada para modelar a *embedding* de uma sentença. Ao considerar ambas as *features*, ponderamos as palavras contidas nas sentenças e a semântica compartilhada entre elas. Como o rótulo de similaridade para o problema em questão é um valor real na escala entre 1 e 5, aplicamos uma técnica de regressão, a Regressão Linear. Os resultados obtidos mostraram que, apesar da *feature* de *embeddings* ter obtido resultados similares ao sistema *baseline*, ao ser combinada à *feature* TF-IDF, apresentou resultados levemente superiores aos obtidos ao ser usada somente a segunda *feature*. Esse foi o trabalho campeão da competição ASSIN 2016 de similaridade semântica pela primeira colocação entre os trabalhos que participaram da tarefa de similaridade textual para português do Brasil e segunda colocação para português de Portugal.

Palavras chave

Similaridade Sentencial, *word embeddings*, Aprendizagem de Máquina

Abstract

In this paper we present a proposal to automatically label the similarity between a pair of sentences and the results obtained on ASSIN 2016 sentence similarity shared-task. Our proposal consists of using a classical feature of bag-of-words, the TF-IDF model; and an emergent feature, obtained from processing word embeddings. The TF-IDF is used to relate texts which share words. Word embeddings are known by capture the syntax and semantics of a word. Following Mikolov et al. (2013a), the sum of embedding vectors can model the meaning of a sentence. Using both features, we are able to capture the words shared between sentences and their semantics. We use linear regression to solve this problem, once the dataset is labeled as real numbers between 1 and 5. Our results are promising. Although the usage of embeddings has not overcome our baseline system, when we combined it with TF-IDF, our system achieved better results than only using TF-IDF. Our results achieved the first collocation of ASSIN 2016 for sentence similarity shared-task applied on brazilian portuguese sentences and second collocation when applying to Portugal portuguese sentences.

Keywords

Sentence Similarity, word embeddings, Machine Learning

1 Introdução

Pesquisas sobre similaridade entre documentos se iniciaram com foco na área de Recuperação de Informação em que, dada uma *query*, retorna os documentos mais similares a ela. A literatura apresenta diferentes abordagens para modelar a similaridade entre documentos. Podemos citar: abordagens por palavras (*bag-of-words*), que calculam a similariade lexical, ou n-grams (Salton, 1989; Damashek, 1995), que conseguem capturar a semântica contida nas sequências de n palavras; e também abordagens mais complexas como *Latent Semantic Analysis* (LSA) (Deerwester et al., 1990; Landauer & Dumais, 1997), que visa calcular a similaridade semântica de todo o documento, e não apenas a lexical.

Entre os trabalhos clássicos da literatura de similaridade de documentos, podemos citar

trabalhos que calcularam a similaridade textual de uma perspectiva matemática, utilizando estatística ou teoria de probabilidade (Ponte & Croft, 1998), trabalhos que utilizam recursos léxicos para calcular a semântica em um parágrafo ou no documento (Rada et al., 1989; Resnik, 1995) e outros trabalhos que combinam todas essas ideias (Rodríguez & Egenhofer, 2003). Esses métodos, no entanto, possuem dificuldades em lidar com a esparsidade de dados, que não proporciona frequência suficiente para métodos probabilísticos nem ocorrência de algumas palavras contidas em recursos lexicais. Portanto, nenhum desses trabalhos é apropriado para lidar com a similaridade sentencial.

Trabalhos subsequentes foram desenvolvidos com o propósito de lidar com a esparsidade de dados na similaridade sentencial (Li et al., 2006; Liu et al., 2007). No entanto, esses trabalhos possuem a deficiência de serem dependentes de córpus ou *wordnet*. Essa dependência restringe um método, muitas vezes, a ser aplicado apenas a uma determinada língua devido à características únicas dessa língua, buscadas em um recurso compilado.

Trabalhos recentes utilizam o conceito de *embeddings* (Mikolov et al., 2013b) para calcular a similaridade entre sentenças, parágrafos e documentos. *Word Embeddings* são modelos preditivos de semântica distribucional que se baseiam em redes neurais, sendo mais recentes que trabalhos clássicos como *Latent Semantic Analysis*, que utiliza *Singular Value Decomposition* (SVD) para fazer matrizes densas (Landauer et al., 1998), ou os clássicos modelos distribucionais descritos e utilizados há 20 anos por Grefenstetti (1996). A vantagem da abordagem por *embeddings*, além da baixa esparsidade de dados, é a independência de recursos léxicos, sintáticos e semânticos. Um modelo de *embeddings* necessita unicamente de um grande córpus de treinamento que, se for apropriado para a tarefa alvo, modelará bem o contexto das palavras e não acarretará na esparsidade de dados. Podemos citar o trabalho de Kenter e de Kenter & de Rijke (2015) que utilizou *word embeddings* para calcular a similaridade semântica entre textos curtos. Os autores treinaram um modelo de *embeddings* utilizando um córpus de 100 bilhões de palavras obtidas do *website* Google News. O gênero jornalístico é comumente utilizado para treinamento de *embeddings* por ser um gênero genérico, o que não limita o modelo treinado à um determinado cenário ou aplicação.

Esse trabalho apresenta uma proposta simples para cálculo da similaridade sentencial. Utilizamos uma *feature* clássica, a TF-IDF (*term frequency–inverse document frequency*), e também uma *feature* emergente, obtida por meio de *word embeddings*. As próximas seções seguem do seguinte modo: na Seção 2, são apresentadas as duas *features* propostas nesse trabalho e também a *baseline*, desenvolvida para validar a eficácia das *features* propostas; na Seção 3, são apresentados os resultados obtidos e uma breve discussão sobre eles; na Seção 4, são descritos alguns trabalhos relacionados, recuperados da SemEval-2014 Task 1, cujo objetivo também foi o cálculo da similaridade sentencial e; na Seção 5, são listadas as conclusões desse trabalho.

2 Features

Nesse trabalho, propomos o uso de duas *features*: uma relacionada com *word embeddings* e outra com o modelo TF-IDF. Também propomos uma *feature baseline* para validar a eficácia das *features* propostas. Nas subseções a seguir, apresentamos as *features* utilizadas nesse trabalho e a motivação para seu uso: na Subseção 2.1, detalhamos a *feature* obtido por meio de *word embeddings*; na Subseção 2.2, detalhamos a *feature* obtida por TF-IDF e, na Subseção 2.3, apresentamos a *feature baseline*.

2.1 Word Embeddings

A abordagem para modelagem de palavras no espaço vetorial utilizada nesse trabalho foi a Skip-Ngram, proposta por Mikolov et al. (2013b). Essa abordagem se baseou nos tradicionais modelos de língua, no entanto, ao invés de utilizar uma sequência de n palavras para predizer a palavra no instante $n+1$, ela utiliza uma única palavra i para predizer a janela j de palavras ao seu redor. Dessa forma, a *embedding* de uma palavra representa o contexto no qual ela ocorre, capturando relações sintáticas e semânticas. Um exemplo clássico da literatura para a língua inglesa mostra que ao subtrair o vetor da *embedding* de *homem* do vetor da *embeddings* de *rei* e somar o vetor da *embeddings* de *mulher*, chega-se a um *embedding* cujo vetor é muito similar ao de *rainha* (Turney, 2006). Com esse exemplo percebemos que a troca do gênero muda o substantivo em si, mas mantém a semântica correta, a versão feminina de *rei*.

Utilizamos o sistema word2vec[1] para a modelagem das *embeddings* por contér o algoritmo

[1]Disponível em `https://code.google.com/archive/p/word2vec/`.

de treinamento Skip-Ngram. O córpus utilizado para treinamento contém 3 bilhões de tokens em português brasileiro, composto por textos do *website* G1, da Wikipédia e do córpus PLN-Br (Bruckschen et al., 2008). Definimos que cada *embedding* seria composta por um vetor de 600 dimensões, tamanho considerado suficiente nos experimentos realizados por Mikolov et al. (2013a). Todas as palavras foram mapeadas para caixa baixa a fim de reduzir esparsidade de dados no córpus. Também definiu-se um mapeamento das palavras com apenas uma ocorrência no córpus para um token genérico *UNK*. Toda nova palavra não encontrada no vocabulário do córpus de treinamento também é mapeada para a *embedding* de *UNK*. É interessante observar que foi possível replicar o exemplo *rei-rainha*, clássico na literatura de *embeddings* da língua inglesa, para o nosso modelo treinado com textos em português brasileiro. Isso reforça que a abordagem de *embeddings* é independente de língua, dependendo apenas do córpus de treinamento.

Para calcularmos a similaridade entre os pares de sentenças, utilizamos o modelo treinado de *word embeddings* para representar as sentenças. O trabalho de Mikolov et al. (2013b) mostra que ao somar os vetores das *embeddings* das palavras de uma sentença, temos como resultado uma *embedding* que representa a sentença. Apesar de não terem sido encontrados trabalhos na literatura que avaliem a qualidade com que a composição de *embeddings* representa uma sentença, intuitivamente percebemos que, se a *embedding* de uma palavra representa o contexto em que ela ocorre, a soma das *embeddings* dessas palavras compõe a soma dos seus contextos. Uma abordagem similar para a tarefa de similaridade textual foi abordada por Bjerva et al. (2014) na SemEval-2014 Task 1. Os autores utilizaram, entre outras *features*, a similaridade do cosseno entre as somas das *embeddings* das sentenças. O sistema desenvolvido pelos autores obteve a terceira melhor colocação na tarefa de similaridade textual da SemEval-2014 Task 1. No âmbito da semântica distribucional composicional, o trabalho de Mitchell & Lapata (2008) obteve melhores resultados ao usar a multiplicação vetorial ao invés da soma. Apesar de termos avaliado ambos os métodos, reportamos apenas os resultados da soma vetorial pois os resultados obtidos foram melhores.

O uso das *embeddings* como *feature* é dado pela similaridade do cosseno entre as *embeddings* dos pares de sentenças. O valor da similaridade entre os dois vetores de *embeddings* é utilizamo como uma *feature* para o sistema de regressão.

2.2 TF-IDF

A fim de utilizar uma abordagem clássica da área de PLN (Processamento de Linguagem Natural) para representação sentencial, realizamos uma modelagem TF-IDF das sentenças do córpus. Sabendo que a modelagem TF-IDF sofre com a esparsidade de dados, utilizamos apenas os *stems* das palavras de conteúdo das sentenças para representá-las, conseguindo dessa forma uma matriz TF-IDF reduzida. Além disso, sabemos que as sentenças a serem avaliadas são curtas e que não necessariamente contém as mesmas palavras. Assim, expandimos o vocabulário das sentenças buscando sinônimos para cada palavra de conteúdo no TEP (Thesaurus para o português do Brasil) (Maziero & Pardo, 2008). Verificamos que, ao expandir os sinônimos para todas as palavras de conteúdo de uma sentença, os vetores TF-IDF das sentenças se tornam muito similares, de forma a não conseguirmos distinguir sentenças similares das distintas. Portanto, empiricamente, limitamos a expansão de sinônimos para palavras de conteúdo que possuem até 2 sinônimos no TEP. Essa decisão foi tomada com base em experimentos no conjunto de treinamento disponibilizado pela comissão organizadora do ASSIN.

O uso do TF-IDF como *feature* é dado pela distância do cosseno entre os vetores TF-IDF dos pares de sentenças. Utilizamos esse valor como uma *feature* para o sistema de regressão.

2.3 Baseline

A fim de avaliar a eficácia das *features* propostas nesse trabalho, elaboramos um *baseline* para avaliação. A *feature baseline* consiste na proporção de palavras compartilhadas entre as duas sentenças. Essa *feature* não captura a semântica latente das sentenças. Por exemplo, mesmo que duas sentenças compartilhem uma quantidade substancal de palavras, um sinal de negação contido em uma dessas sentenças pode inverter o seu significado em relação a outra sentença. Assim, as *features* propostas são eficazes se capturarem informações latentes sobre as sentenças, de forma a proporcionar uma melhor performance ao sistema que automatiza a similaridade sentencial.

3 Experimentos

Nós treinamos 2 sistemas de Regressão Linear com os conjuntos de treinamento compostos por pares de sentença em português do Brasil (PTBR) e em português de Portugal (PTPT) disponibilizados pela comissão organizadora do

ASSIN. Ambos os conjuntos contém 3,000 pares de sentenças cada. Cada sistema foi treinado com variação de *features*: utilizando a *feature baseline*; utilizando apenas *embeddings*; utilizando apenas *TF-IDF*; e uma versão utilizando *embeddings* e TF-IDF. Avaliamos as versões PTBR do nosso sistema sobre o conjunto de teste disponibilizado na *shared-task*, composto por 2,000 pares de sentenças em PTBR. Analogamente, avaliamos as versões PTPT do nosso sistema sobre o conjunto de testes PTPT da *shared-task*. Utilizamos as medidas Correlação de Pearson (CP) e Erro Quadrado Médio (EQM) para avaliar a qualidade das *features* propostas na tarefa de similaridade sentencial via método de regressão.

	PT-BR		PT-PT	
Feature	CP	EQM	CP	EQM
Baseline	0,57	0,50	0,60	0,49
Embeddings	0,58	0,50	0,55	0,83
TF-IDF	0,68	0,41	0,70	**0,39**
Embeddings + TF-IDF	**0,70**	**0,38**	**0,70**	0,66

Tabela 1: Avaliação das *features* propostas para cálculo de similaridade sentencial, utilizando Regressão Linear, nos conjuntos de teste da ASSIN *shared-task*.

Verificando os resultados apresentados na Tabela 1, percebemos que o uso apenas da *feature* obtida das *word embeddings* não resultou em uma boa performance da Regressão Linear. Entendemos que, apesar da literatura apontar que a soma das *embeddings* de uma sequência de palavras representar a sintaxe-semântica dessa sequência, essa representação se torna genérica, não representando de fato a informação ali contida. Também devemos ponderar que, como o modelo de *embeddings* foi gerado sobre textos em PTBR, ele não está calibrado para lidar com a variante da língua PTPT – o que justifica o aumento de EQM na avaliação sobre o conjunto PTPT ao adicionar a *feature Embeddings* à TF-IDF. Além disso, a soma das *embeddings* pode não ser a melhor forma de manipular essa informação. O trabalho de Gabrilovich & Markovitch (2007) propõe o ponderamento das *embeddings* das palavras de um documento pela frequência com que essas palavras aparecem na língua. O trabalho de Yuan et al. (2016) mostra que o uso dessa modelagem melhora a performance da tarefa de desambiguização lexical de sentidos ao utilizar redes neurais.

Os resultados também nos mostram que o uso da *feature* TF-IDF resultou em uma performance significativa da Regressão Linear em relação ao uso da *feature baseline*. É interessante observar que a representação TF-IDF segue o modelo *bag-of-words*, que implica a perda da ordem das palavras e na semântica latente. Não podemos afirmar que o resultado final do nosso sistema, que utiliza ambas as *features*, é superior ao do sistema que utiliza apenas TF-IDF, devido a falta de um teste de significância estatística. No entanto, especulamos que o uso das *embeddings* contribui para que o sistema capture a semântica da sentença em casos em que o significado do contexto importa, cenário em que o TF-IDF é insuficiente.

Os resultados obtidos pelo sistema desenvolvido nesse trabalho obtiveram primeiro lugar entre os competidores ao aplicar o sistema no córpus PTBR e segundo lugar ao aplicar o sistema no córpus PTPT. No caso geral, ao unir os córpus PTBR e PTPT, nós fomos os melhores colocados, com **0,68** de CP e **0,52** de EQM.

4 Trabalhos Relacionados

O SemEval 2014 disponibilizou uma *shared-task* (SemEval-2014 Task 1)[2], cujo um dos objetivos foi calcular a similaridade sentencial de um par de sentenças. Foi disponibilizado um dataset, o SICK, que contém 10,000 pares de sentenças, sendo 5,000 pares para treinamento e 5,000 pares para teste. Essa *shared-task* inspirou a organização da ASSIN, competição com propósito similar cujo foco voltou-se para a língua portuguesa. Nessa seção serão listados três trabalhos do SemEval-2014 Task 1 que trataram de similaridade sentencial.

O trabalho de Zhao et al. (2014) considerou um vasto conjunto de *features*. Entre as *features* utilizadas, podemos citar: tamanho de sentenças, similaridade superficial (distância do cosseno), similaridade semântica, *ngrams* com base em córpus de referência, entre outras. Esse trabalho foi o primeiro colocado para a tarefa de similaridade sentencial, obtendo 0,828 de CP e 0,325 de EQM.

O trabalho de Bjerva et al. (2014) utilizou uma variedade de *features*, das quais podemos citar: tamanho das sentenças, substantivos e verbos compartilhados entre as sentenças, diferenças entre os conceitos Wordnet das palavras das sentenças e distância do cosseno das *word embeddings* das sentenças. Esse trabalho foi o terceiro colocado para a tarefa de similaridade sentencial, obtendo 0,827 de CP e 0,322 de EQM.

O trabalho de Lai & Hockenmaier (2014) utiliza *features* que consideram a proporção de palavras compartilhadas entre as sentenças, alinha-

[2]Anais disponíveis em `http://www.aclweb.org/anthology/S/S14/S14-2.pdf#page=349`.

mento entre as sentenças, presença de negação e a similaridade semântica entre o conjunto de palavras não compartilhado entre as sentenças. Esse trabalho foi o quinto colocado para a tarefa de similaridade sentencial, com 0,799 de CP e 0,369 de EQM.

5 Conclusão

Esse artigo apresentou os resultados obtidos pela equipe *Solo Queue* na tarefa de similaridade textual da ASSIN 2016 *shared-task*. Nossa proposta consiste no uso de uma *feature* clássica da classe *bag-of-words*, a TF-IDF; e uma *feature* emergente, obtida por meio de *word embeddings*. Sabemos que a medida TF-IDF pode ser utilizada para relacionar documentos que compartilham palavras e, portanto, pode ser utilizada para relacionar sentenças. *Word embeddings* são conhecidas por modelar o contexto das palavras e podem ser utilizadas para modelar o contexto de uma sentença. Nossa equipe obteve os melhores resultados da *shared-task* ao avaliar o sistema desenvolvido sobre o conjunto de teste de pares de sentença em português do Brasil e segundo lugar ao avaliar sobre o conjunto de teste de pares de sentença em português de Portugal. No caso geral de avaliação, em que juntou-se os córpus, nosso grupo foi o melhor colocado. Acreditamos que melhores resultados podem ser obtidos ao investigar-se uma melhor ponderação das *embeddings* das palavras para modelar a *embedding* de sua sentença, como apresentado por Gabrilovich & Markovitch (2007) e Yuan et al. (2016). Ainda assim, a composição das *embeddings* de uma sequência de palavras não mantém a ordem delas, perdendo parte da semântica contida na sentença. Para resolver esse problema, vale avaliar o uso de uma rede LSTM para modelar a *embedding* de uma sentença a partir das *embeddings* das palavras dessa sentença. Redes LSTM são conhecidas por manterem a ordem de entrada dos elementos (Hochreiter & Schmidhuber, 1997). Também sabemos que o fato do nosso conjunto de *embeddings* ter sido treinado apenas sobre textos em Português do Brasil desafiou o sistema a lidar com textos em Português de Portugal. Assim, o treinamento de um modelo de *embeddings* que contemple ambas as línguas é o ideal pois, apesar das línguas compartilharem muitas características, suas nuances geram desafios particulares que merecem atenção.

Agradecimentos

Agradecemos ao aporte financeiro da FAPESP (p. 2016/00500-1) que financia esse projeto de pesquisa.

Referências

Bjerva, Johannes, Johan Bos, Rob van der Goot & Malvina Nissim. 2014. The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. Em *SemEval 2014: International Workshop on Semantic Evaluation*, 642–646.

Bruckschen, M., F. Muniz, J. Souza, J. Fuchs, K. Infante, M. Muniz, P. Gonçalves, R. Vieira & S. Aluísio. 2008. Anotação Lingüística em XML do Corpus PLN-BR. NILC–TR–09–08. Relatório técnico. University of São Paulo.

Damashek, Marc. 1995. Gauging similarity with n-grams: Language-independent categorization of text. *Science* 267(5199). 843–848.

Deerwester, Scott, Susan T Dumais, George W Furnas, Thomas K Landauer & Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41(6). 391–407.

Gabrilovich, Evgeniy & Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. Em *IJCAI*, vol. 7, 1606–1611.

Grefenstetti, Gregory. 1996. Evaluation techniques for automatic semantic extraction: comparing syntactic and window based approaches. Em *Corpus processing for lexical acquisition*, MIT Press.

Hochreiter, Sepp & Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8). 1735–1780.

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

Lai, Alice & Julia Hockenmaier. 2014. Illinois-lh: A denotational and distributional approach to semantics. Em *Proceedings of the 8th International Workshop on Semantic Evaluation*, 329–334.

Landauer, Thomas K. & Susan T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review* 104(2). 211.

Landauer, Thomas K, Peter W Foltz & Darrell Laham. 1998. An introduction to latent semantic analysis. *Discourse processes* 25(2-3). 259–284.

Li, Yuhua, David McLean, Zuhair A Bandar, James D O'shea & Keeley Crockett. 2006. Sentence similarity based on semantic nets and corpus statistics. *Knowledge and Data Engineering, IEEE Transactions on* 18(8). 1138–1150.

Liu, Xiaoying, Yiming Zhou & Ruoshi Zheng. 2007. Sentence similarity based on dynamic time warping. Em *Semantic Computing, 2007. ICSC 2007. International Conference on*, 250–256. IEEE.

Maziero, Erick & Thiago Pardo. 2008. Interface de Acesso ao TeP 2.0 - Thesaurus para o português do Brasil. Relatório técnico. University of São Paulo.

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

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado & Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. Em *Advances in neural information processing systems*, 3111–3119.

Mitchell, Jeff & Mirella Lapata. 2008. Vector-based models of semantic composition. Em *ACL*, 236–244.

Ponte, Jay M & W Bruce Croft. 1998. A language modeling approach to information retrieval. Em *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 275–281. ACM.

Rada, Roy, Hafedh Mili, Ellen Bicknell & Maria Blettner. 1989. Development and application of a metric on semantic nets. *Systems, Man and Cybernetics, IEEE Transactions on* 19(1). 17–30.

Resnik, Philip. 1995. Using information content to evaluate semantic similarity in a taxonomy. arXiv preprint @ cmp-lg/9511007.

Rodríguez, M Andrea & Max J Egenhofer. 2003. Determining semantic similarity among entity classes from different ontologies. *Knowledge and Data Engineering, IEEE Transactions on* 15(2). 442–456.

Salton, Gerard. 1989. *The transformation, analysis, and retrieval of automatic text processing.* Reading: Addison-Wesley.

Turney, Peter D. 2006. Similarity of semantic relations. *Computational Linguistics* 32(3). 379–416.

Yuan, Dayu, Ryan Doherty, Julian Richardson, Colin Evans & Eric Altendorf. 2016. Word sense disambiguation with neural language models. arXiv preprint @ arXiv:1603.07012.

Zhao, Jiang, Tian Tian Zhu & Man Lan. 2014. Ecnu: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. Em *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 271–277.

http://www.linguamatica.com/

linguaMATICA

Avaliação de Similaridade Semântica e de Inferência Textual