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.