Improving distributed vector representation of short and noisy texts in the context of online classification

Renato M. Silva[1], **Johannes V. Lochter**[2,3], **Tiago A. Almeida**[1]

[1]Departamento de Computação (DComp)
Universidade Federal de São Carlos (UFSCar), Sorocaba, SP, Brasil

[2]Departamento de Sistemas e Energia (DSE)
Universidade Estadual de Campinas (UNICAMP), Campinas, SP, Brasil

[3]Departamento de Engenharia da Computação
Centro Universitário Facens, Sorocaba, SP, Brasil

renatoms@dt.fee.unicamp.br, johannes.lochter@facens.br,
talmeida@ufscar.br

Abstract. *The classification of messages generated by users on social networks and other Internet platforms is challenging because they are often short and full of slang, abbreviations, and idioms, which hinders the feature extraction. To address this problem, this study proposes a data augmentation technique to increase the number of data in order to improve the quality of the textual representation model and, consequently, the performance in the classification. The proposed technique is evaluated in the online classification of sentiment in Twitter messages. The experiments were carefully performed and statistical analysis of the results indicated that the data augmentation is effective in online classification of short and noisy text messages.*

Resumo. *A classificação de mensagens geradas pelos usuários em redes sociais e outras plataformas da Internet é desafiadora porque costumam ser curtas e repletas de gírias, abreviações e expressões idiomáticas, o que dificulta a extração dos atributos. Este trabalho propõe uma técnica de expansão de dados para aumentar o número de amostras com o objetivo de melhorar a qualidade do modelo de representação textual e elevar o desempenho na classificação. A técnica proposta é avaliada em um cenário de classificação online de sentimento em mensagens do Twitter. Os experimentos foram diligentemente realizados e uma análise estatística dos resultados indicou que a expansão de dados é efetiva na classificação online de mensagens de texto curtas e ruidosas.*

1. Introdução

A mineração de texto é uma tarefa que vem se tornando cada vez mais importante nos dias atuais. Usuários da Internet têm gerado um grande volume de textos nas redes sociais, *blogs*, sites de notícias, comércios eletrônicos e aplicativos de *smartphones*, por meio de comentários e mensagens instantâneas. Esses textos têm sido usados para automatizar diversas tarefas de grande relevância nos dias de hoje como filtragem de *spam*, identificação de comentários racistas e detecção de autoria.

Outra tarefa que envolve textos produzidos por usuários da Internet com grande relevância atualmente é a mineração de opiniões, também conhecida como análise de sentimento [Socher 2015]. Essa aplicação tem ganhado cada vez mais destaque porque as mensagens geradas pelos usuários são fontes valiosas para as empresas terem uma espécie de termômetro da aceitação das suas mercadorias e serviços. Da mesma forma, muitos consumidores costumam utilizar as opiniões de outros usuários para decidir se devem adquirir um bem ou serviço [Das et al. 2017].

A mineração das mensagens geradas pelos usuários da Internet é desafiadora, pois muitas vezes elas possuem um baixo limite de caracteres e são repletas de

ruídos, como palavras grafadas incorretamente, abreviações, gírias e *emojis*. Um ambiente especialmente rico nesse tipo de conteúdo, conhecido como texto curto e ruidoso [Lochter et al. 2016], é o microblog Twitter. Essa plataforma já registrou um recorde de 143 mil mensagens (também conhecidas como *tweets*) por segundo, sendo que mais de 500 milhões de mensagens são postadas diariamente[1].

Neste contexto, como gírias e abreviações novas surgem a todo instante, os modelos de classificação geralmente empregados na literatura precisam ser constantemente retreinados, pois o processo de aprendizado costuma ser *offline*, o que requer que todos os exemplos de treinamento sejam apresentados de uma única vez. O retreinamento dos modelos costuma ser custoso e a medida que a base rotulada cresce, demanda cada vez mais tempo e recursos computacionais.

Devido ao número crescente de mensagens curtas e ruidosas, um modelo de classificação ideal deveria ser capaz de aprender de forma contínua com a chegada de novas amostras para evitar o retreinamento frequente. Métodos de classificação com esse tipo de treinamento, também conhecido como treinamento *online*, são mais apropriados para problemas dinâmicos e de larga escala [Silva et al. 2017, Losing et al. 2018]. No entanto, a maioria dos trabalhos na literatura não trata as tarefas de classificação de textos curtos e ruidosos como um processo incremental, necessário para se adequar ao cenário moderno e inflado de dados.

Outro aspecto importante diz respeito à representação computacional do texto. A representação *bag of words*, uma das mais tradicionais da literatura, apresenta uma série de deficiências bem conhecidas, como a alta esparsidade e dimensionalidade, perda de localização dos termos e ambiguidade [Lochter et al. 2016]. Devido a essas deficiências, nos últimos anos, foram propostos vários métodos para substituí-la. Entre eles, os modelos de representação distribuída têm se destacado e estão se tornando o estado-da-arte na representação computacional de textos. O mapeamento da representação distribuída é denso, ao contrário da representação esparsa da *bag of words*, tem baixa dimensionalidade e representa melhor a relação semântica entre os termos [Mikolov et al. 2011, Hirschberg e Manning 2015].

Um bom modelo de representação distribuída geralmente precisa ser treinado com uma grande quantidade de dados. Por isso, geralmente são empregados grandes corpos de textos bem escritos e generalistas para treinar os modelos usados na representação textual das amostras, tais como corpos de notícias (*e.g.*, Google News) e enciclopédias (*e.g.*, Wikipedia). Porém, [Lochter et al. 2018] recomendam que o corpo de texto usado para treinar o modelo de representação distribuída seja composto por dados com características semelhantes às amostras da aplicação. Os autores mostraram que em problemas de classificação de textos curtos e ruidosos, os métodos de classificação obtiveram melhor desempenho quando os vetores de atributos foram obtidos por um modelo de representação distribuída treinado com uma base de dados também composta por textos curtos e ruidosos ao invés de textos formais.

Diferentes representações podem ser geradas a partir de diferentes técnicas de representação ou diferentes corpos de texto. Modelos de representação distribuída treinados com diferentes corpos de texto podem capturar padrões semânticos distintos e complementares. Neste sentido, alguns trabalhos recentes na literatura propõem realizar a combinação dos vetores de atributos gerados por diferentes modelos [Goikoetxea et al. 2016, Ghannay et al. 2016]. Com base nas conclusões reportadas por [Lochter et al. 2018], é razoável presumir que, em tarefas de processamento de textos, os próprios documentos da aplicação são aqueles que oferecem a melhor relação sintática e semântica dos dados. Portanto, combinar modelos genéricos de representação com um modelo específico, treinado com os próprios documentos da aplicação, poderá resultar em uma representação generalista e ao mesmo tempo capaz de capturar as particularidades específicas do domínio do problema. Para tanto, idealmente esse modelo específico

[1] *Twitter Engineering Blog*. Disponível em https://blog.twitter.com/engineering/en_us/a/2013/new-tweets-per-second-record-and-how.html. Data de acesso: 28/06/2019.

de representação deveria ser treinado com os próprios dados de treinamento da tarefa de classificação. Contudo, muitas vezes não é viável obter uma grande base de dados com textos da aplicação, pois pode demandar a intervenção de um especialista. Esse problema é agravado em cenários de classificação *online*, pois o número de amostras da aplicação inicialmente disponível, em geral, é muito pequeno [Silva et al. 2017].

Diante disso, com o objetivo de aumentar o número de amostras da aplicação e minimizar os problemas causados por amostras curtas e ruidosas, este trabalho propõe um método de expansão de dados (*data augmentation*) que emprega técnicas de normalização léxica e indexação semântica. Esse método é usado para tratar os termos ruidosos e aumentar o número de amostras disponíveis para o treinamento dos modelos de representação distribuída.

A hipótese desta pesquisa é que a técnica de expansão de dados é capaz de gerar um grande volume de amostras semanticamente relacionadas com os dados originais. Desta forma, o modelo de representação, ao ser treinado com esses dados, consegue aprender as diferentes representações para uma mesma amostra (*i.e.*, palavras que a compõem) e associá-las, evitando que novas amostras precisem passar pela etapa de normalização e indexação, pois ele será capaz de lidar com as amostras em seu estado original. Por fim, esse modelo específico, quando combinado a um modelo genérico treinado com um grande corpo de texto, oferecerá uma representação mais robusta, generalista e, ao mesmo tempo, fiel aos documentos da aplicação, que será refletido na melhoria do desempenho na tarefa de classificação.

O restante deste artigo está organizado nas seguintes seções: a Seção 2 aborda os métodos de representação distribuída textual utilizados neste trabalho; a Seção 3 apresenta a técnica de expansão semântica das amostras; a Seção 4 detalha o protocolo experimental seguido para obter os resultados descritos na Seção 5. Finalmente, as principais conclusões e trabalhos futuros são discutidos na Seção 6.

2. Representação distribuída de textos

Estudos na última década levaram à criação de modelos de representação distribuída, as quais vêm demonstrando serem mais eficientes que a tradicional técnica de representação distributiva *bag of words*. A representação distribuída de textos tornou-se estado-da-arte em muitas tarefas de processamento de linguagem natural, como tradução de máquina [Vaswani et al. 2017], análise de sentimentos [Xie et al. 2019], entre outras.

O bom desempenho da representação distribuída se deve à capacidade de atribuir a cada palavra, ou *token*, um vetor de tamanho fixo com características semânticas ou sintáticas mapeadas a partir de treinamento não-supervisionado em um corpo de texto. Na literatura, tais representações são induzidas por várias técnicas, sendo a modelagem baseada em redes neurais artificiais, como Word2Vec [Mikolov et al. 2013] e FastText [Bojanowski et al. 2017], a mais empregada.

2.1. Word2Vec

Word2Vec é um método de representação baseado em rede neural que processa corpos de texto para associar termos que aparecem em contextos semelhantes em um mesmo espaço vetorial. Essa associação acontece por meio de similaridade semântica e sintática obtidas ao analisar um corpo de texto utilizando os seguintes algoritmos: *continuous bag of words* (CBOW) e Skip-Gram [Mikolov et al. 2013].

No algoritmo CBOW, o vocabulário é coletado no corpo de texto e cada entrada do vocabulário é associada a um vetor de tamanho fixo inicializado aleatoriamente em uma matriz. Em seguida, o corpo de texto é processado ao deslizar uma janela de contexto por toda sua extensão. A janela de contexto fornece o alvo da rede e os termos de contexto. O termo central da janela é removido e utilizado como alvo da rede, enquanto os termos restantes são substituídos por seus respectivos vetores a partir da matriz. Conforme os vetores de contexto não predizem corretamente a palavra alvo, seus valores são ajustados.

O algoritmo Skip-Gram é muito semelhante ao CBOW, mas a separação da janela de contexto é feita de forma diferente. O termo central é colocado na entrada da rede e o contexto é colocado na saída. Desse modo, a rede ajusta os vetores de contexto a medida que o termo central não prevê corretamente o contexto (saída da rede).

O resultado final para ambos os algoritmos são os vetores, ou *word embeddings*, ajustados ao processamento do corpo de texto, o qual captura características sintáticas e semânticas com base nos contextos em que os termos co-ocorrem.

2.2. FastText

FastText é uma rede neural similar à Word2Vec e pode ser interpretada como uma extensão da mesma. A principal diferença com relação a essas duas redes é a capacidade do método FastText encontrar representações para subpalavras, ou unidades morfológicas menores que o próprio termo, tais como prefixos, radicais e sufixos [Bojanowski et al. 2017].

FastText pode operar no nível de caractere do mesmo modo que Word2Vec opera no nível de palavras, portanto é capaz de encontrar representações distribuídas de partes da palavra e combiná-las para formar palavras ausentes do vocabulário. Essa vantagem torna o modelo menos sensível ao fenômeno das palavras fora de vocabulário. Este fenômeno é muito comum em tarefas de processamento de linguagem natural e acontece quando uma palavra que não foi vista no treinamento aparece em alguma amostra da partição de teste. Neste caso, não existe uma representação conhecida para a palavra em questão.

3. Expansão semântica de dados

Uma das características das aplicações que exigem classificação *online* de mensagens curtas e ruidosas, é que o número de amostras inicialmente disponíveis geralmente é pequeno [Silva et al. 2017]. Por isso, pode não haver uma quantidade suficiente de amostras com as mesmas características do domínio da aplicação para treinar adequadamente os modelos de representação distribuída. Uma das formas de aproveitar melhor os dados é criar novas representações deles por meio de expansão de dados [Saito et al. 2017].

Neste artigo, é proposta uma técnica automática de expansão semântica de dados que usa as informações semânticas das mensagens para criar novas mensagens similares. Essa técnica aumenta o número de dados disponíveis podendo melhorar os modelos de representação distribuída. Para isso, ela utiliza os três métodos de processamento de linguagem natural apresentados a seguir.

- *Normalização léxica*: consiste em traduzir gírias, expressões idiomáticas e abreviações de palavras geralmente usadas pelos usuários da Internet para a sua forma canônica.
- *Indexação semântica*: técnica de extração de informações semânticas que obtém diferentes significados para um determinado termo. O uso dessa técnica pode aumentar a quantidade de termos de um texto, sendo considerada uma técnica de geração de conceitos, expansão ou enriquecimento textual [Lochter et al. 2016]. Para a geração de conceitos, geralmente é utilizado algum repositório semântico, como o `LDB BabelNet`[2]. Porém, outras fontes de informação podem ser utilizadas, como a Wikipedia [Lochter et al. 2016, Vo e Ock 2015].
- *Desambiguação*: técnica usada para selecionar os conceitos mais relevantes de um termo de acordo com o contexto do texto. Em algumas aplicações, tais como a proposta de [Lochter et al. 2016], a desambiguação é considerada uma etapa posterior à indexação semântica, pois primeiramente são encontrados diferentes significados para o termo e depois são selecionados os significados mais relevantes ao contexto.

[2]O `LDB BabelNet` está disponível em: `http://babelnet.org/`. Data de acesso: 01/07/2019.

A técnica de expansão semântica de dados proposta neste trabalho está ilustrada na Figura 1. Na primeira etapa, a normalização léxica é usada para diminuir o ruído das mensagens trocando os termos grafados incorretamente por variantes corretas presentes em dicionários léxicos. Para isso, é empregado um dicionário Lingo compilado a partir de diversas fontes e um dicionário de inglês contendo 466 mil palavras. Ambos dicionários estão disponíveis respectivamente em `https://github.com/jlochter/semantic-data-augmentation` e `https://github.com/dwyl/english-words`.

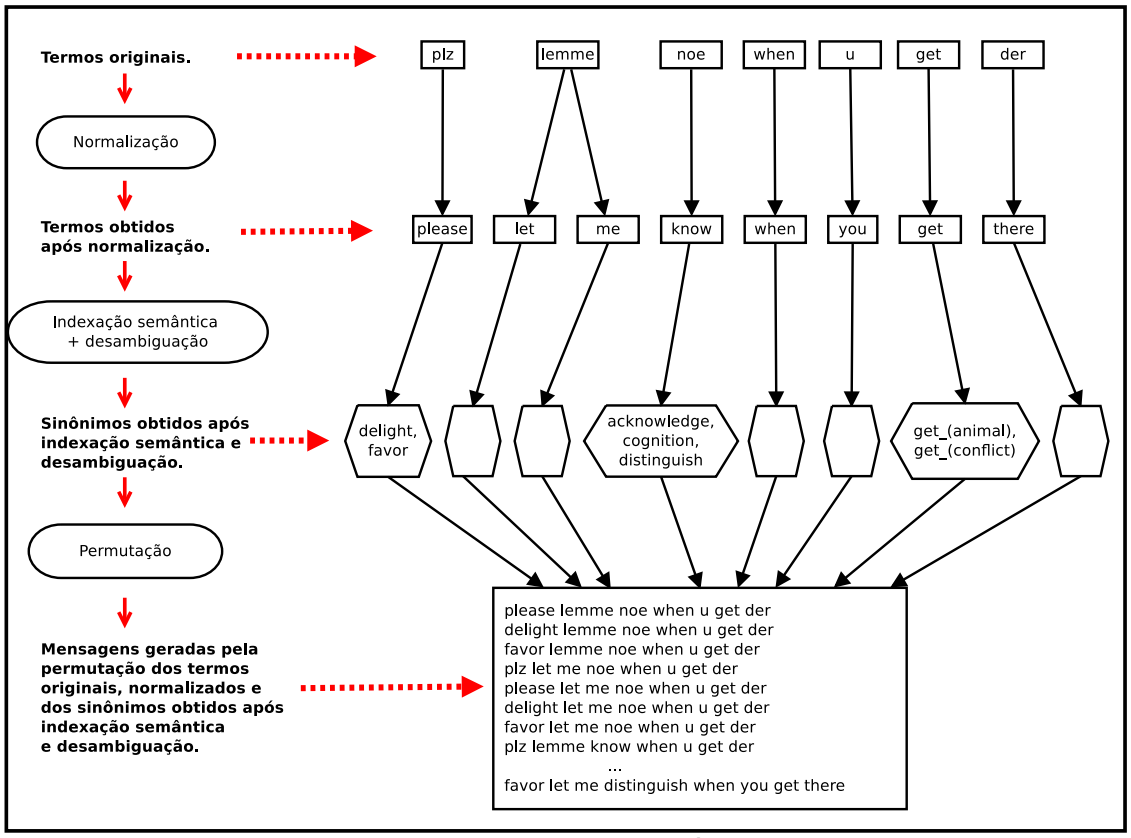


Figura 1. Exemplo da expansão semântica de dados.

Em seguida, é empregada a indexação semântica para representar cada termo da mensagem como um conjunto de sinônimos e, então, o processo de desambiguação seleciona somente os termos do conjunto de sinônimos que são pertinentes ao contexto da mensagem. Nesta etapa, as amostras são apresentadas através de API ao serviço Babelfy [Moro et al. 2014], que retorna um arquivo estruturado identificando os termos que puderam ser desambiguados e seus respectivos sinônimos.

No próximo passo, a técnica proposta executa a permutação entre os termos originais da mensagem, os termos normalizados e os termos do conjunto de sinônimos selecionados no processo de desambiguação. O resultado da permutação é um conjunto novo de amostras, onde cada amostra é composta por um termo de cada conjunto. Cada nova amostra corresponde a uma variante sintática e morfológica com mesma semântica da amostra original que a criou. A técnica de expansão semântica de dados foi implementada em Python 3 e seu código fonte está disponível publicamente na plataforma Github[3].

Após o emprego dessa técnica, é esperado que seja viável realizar o treinamento de um bom modelo de representação a partir dos dados da própria aplicação, que estão disponíveis na partição de treinamento da tarefa de classificação *online*. O método de

[3]*Semantic Data Augmentation*. Disponível em: `https://github.com/jlochter/semantic-data-augmentation`. Data de acesso: 01/07/2019.

expansão aqui proposto é capaz de produzir uma grande quantidade de amostras semanticamente equivalentes às amostras originais disponíveis inicialmente. Assim, o modelo de representação pode aprender as diferentes representações de um mesmo dado e associá-las semanticamente. É também esperado que, com isso, o modelo de representação seja capaz de lidar com novos dados em seu estado original, evitando que novas amostras coletadas no processo *online* também precisem passar pelas etapas de normalização e indexação semântica. Por fim, é previsto que esse modelo de representação, quando combinado com outro modelo mais genérico, resulte em uma representação mais robusta e ao mesmo tempo generalista, que será refletida na melhoria do desempenho final da capacidade preditiva dos métodos de classificação.

4. Protocolo experimental

Para avaliar a hipótese deste trabalho, foi considerado o cenário de classificação *online* da polaridade de sentimento em mensagens do Twitter. Essa tarefa envolve analisar uma pequena base de amostras previamente rotuladas, produzindo um modelo capaz de classificar as novas amostras com relação ao sentimento que elas expressam.

Para dar credibilidade aos resultados e tornar os experimentos reprodutíveis, todos os testes foram realizados com as seguintes bases de dados clássicas, reais e públicas: Archeage, Hobbit e IPhone6 criadas por [Lochter et al. 2016], além da OMD (Obama-McCain *debate*) [Shamma et al. 2009], HCR (*health care reform*) [Speriosu et al. 2011], Sanders[4] [Speriosu et al. 2011], SS-Tweet (*sentiment strength* Twitter *dataset*) [Thelwall et al. 2012], STS-Test (Stanford Twitter *sentiment test set*) [Agarwal et al. 2011] e UMICH[5] [Lochter et al. 2016].

Em todos os experimentos, os textos foram convertidos para letras minúsculas. Em seguida, a ferramenta de processamento de textos Ekphrasis [Baziotis et al. 2017], especializada em textos de redes sociais, foi usada para normalizar os *emojis*, endereços de páginas da Internet, endereços de email, números, porcentagens, valores monetários, números de telefone e datas. Essa ferramenta também foi empregada para tratar *hashtags* e fazer a tokenização das mensagens.

A Tabela 1 apresenta as principais estatísticas sobre as bases de dados originais. Com o propósito de ilustrar a capacidade de geração de amostras e enriquecimento textual da técnica de expansão semântica, ela foi aplicada para expandir todas as amostras das bases de dados. A Tabela 2 apresenta as estatísticas das bases resultantes após esse processo de expansão. Nas duas tabelas, $|D|$ é o número de documentos, $|D|_{\text{Pos}}$ e $|D|_{\text{Neg.}}$ são, respectivamente, o número de mensagens com polaridade positiva e negativa. $|V|$ é a quantidade de termos únicos na base (tamanho do vocabulário), $\mathcal{M}^t$ e $\mathcal{I}^t$ são, respectivamente, a mediana e a amplitude interquartil (IQR – *interquartile range*) do número de termos por mensagem. Na Tabela 2, $\mathcal{M}^s$ e $\mathcal{I}^s$ são, respectivamente, a mediana e o IQR do número de mensagens expandidas, geradas a partir de cada mensagem original.

Pelos valores apresentados na coluna $\mathcal{M}^t$, é possível constatar que a quantidade de termos por mensagem é maior nas bases de dados expandidas (Tabela 2). Portanto, a expansão semântica das bases de dados contribui tanto para aumentar o número de amostras (entre 50 e 93 vezes), como para incrementar o tamanho das mensagens. Além disso, o tamanho do vocabulário das bases de dados aumentou, em média, 60% após aplicar a expansão semântica.

4.1. Aprendizado *online*

Para simular um cenário de aprendizado *online*, apenas uma pequena quantidade de dados (30% da base) foi usada para o treinamento inicial dos classificadores. No estágio de teste,

[4]Niek Sanders. Twitter Sentiment Corpus. Sanders Analytics. Disponível em: `http://www.sananalytics.com/lab/twitter-sentiment/`. Data de acesso: 01/07/2019.

[5]UMICH SI650 - Sentiment Classification. Disponível em: `https://www.kaggle.com/c/si650winter11/`. Data de acesso: 01/07/2019.

Tabela 1. Estatísticas das bases de dados originais.

Dataset	D	$\lvert D\rvert_{\text{Pos.}}$	$\lvert D\rvert_{\text{Neg.}}$	$\lvert V\rvert$	$\mathcal{M}^t$	$\mathcal{I}^t$
Archeage	1.520	591	929	2.952	17	12
HCR	1.406	534	872	4.076	25	9
Hobbit	488	327	161	1.302	15	12
IPhone6	522	365	157	1.548	15	14
OMD	1.886	702	1.184	3.835	17	9
Sanders	1.078	515	563	3.027	20	11
SS-Tweet	2.288	1.252	1.036	6.782	17	10
STS-Test	359	182	177	1.580	14	12
UMICH	1.116	617	499	2.129	10	9

Tabela 2. Estatísticas das bases de dados após a expansão.

Dataset	D	$\lvert D\rvert_{\text{Pos.}}$	$\lvert D\rvert_{\text{Neg.}}$	$\lvert V\rvert$	$\mathcal{M}^t$	$\mathcal{I}^t$	$\mathcal{M}^s$	$\mathcal{I}^s$
Archeage	141.473	42.727	98.746	4.810	23	7	63	189
HCR	153.915	55.639	98.276	6.385	28	7	107	177
Hobbit	30.017	17.927	12.090	2.040	23	7	23	138
IPhone6	30.910	17.458	13.452	2.518	25	9	17	138
OMD	171.578	55.814	115.764	6.013	22	8	71	184
Sanders	98.610	39.819	58.791	4.918	25	7	71	188
SS-Tweet	200.127	97.540	102.587	9.677	23	7	56	188
STS-Test	24.906	10.707	14.199	2.589	23	9	31	184
UMICH	55.740	27.719	28.021	3.488	21	12	15	68

foi adotado o método *prequential* que é amplamente empregado para avaliar técnicas de aprendizado *online* [Gama et al. 2013]. Neste método, uma mensagem do conjunto de teste é apresentada por vez ao classificador, que emite a predição. Em seguida, o classificador recebe *feedback* e atualiza o modelo de predição com o rótulo correto.

Foram realizados experimentos com os seguintes métodos tradicionais de aprendizado *online*: *naive* Bayes multinomial (M.NB) [McCallum e Nigam 1998], *naive* Bayes Bernoulli (B.NB) [McCallum e Nigam 1998], Perceptron [Freund e Schapire 1999] e gradiente descendente estocástico (SGD – *stochastic gradient descent*) [Zhang 2004]. Todos os métodos foram implementados em `Python` usando funções da biblioteca `scikit-learn` [Pedregosa et al. 2011]. Para comparar os resultados, foi empregada a medida de desempenho macro F-medida.

4.2. Cenários experimentais

Em todos os experimentos, para o treinamento do modelo de representação genérico, foi utilizado o corpo de texto Twitter7 (T7) [Yang e Leskovec 2011], que contém *tweets* publicados de junho de 2009 a dezembro de 2009. Os *retweets* e mensagens vazias foram eliminados e, em seguida, foram selecionados apenas os *tweets* em língua inglesa, totalizando 364.025.273 *tweets* que foram usados para treinar o modelo de *embeddings*. Os documentos desse corpo de texto foram pré-processados da mesma forma que as amostras das bases de dados apresentadas na Tabela 1.

Foram usadas as técnicas Word2Vec e FastText para treinar essas *word embeddings*. Para ambas, foram gerados vetores com 200 dimensões treinados com o modelo Skip-Gram [Mikolov et al. 2013, Bojanowski et al. 2017]. A função de composição escolhida para representar cada documento como um vetor de tamanho fixo foi a média dos vetores de cada palavra do documento.

Para verificar se a expansão semântica de dados viabiliza o treinamento de um modelo de representação a partir dos dados da aplicação, foram analisados os seguintes cenários:

- **T7**: apenas as *word embeddings* genéricas, treinadas com o corpo de texto T7, foram utilizadas na função de composição de cada amostra da aplicação.
- **T7 + aplicação**: além das *word embeddings* genéricas, treinadas com o corpo de texto T7, outro modelo específico foi treinado com as amostras originais do conjunto de treinamento da aplicação. Nesse cenário, cada amostra foi representada pela concatenação de duas funções de composição. A primeira, utilizou a média

dos vetores obtidos nas *word embeddings* treinadas na T7, e a segunda função utilizou a média das *word embeddings* treinadas nas amostras de treinamento da aplicação.

- **T7 + aplicação expandida**: além das *word embeddings* genéricas, treinadas com o corpo de texto T7, outro modelo específico foi treinado com as amostras expandidas do conjunto de treinamento da aplicação. Cada amostra também foi representada pela concatenação de duas funções de composição. A primeira função considerou a média dos vetores nas *word embeddings* treinadas na base T7, e a segunda função considerou a média dos vetores das *word embeddings* treinadas nas amostras de treinamento expandidas.

A hipótese considerada na escolha desses cenários é que, em aplicações que exigem treinamento *online*, um modelo de *embeddings* treinado apenas com os documentos de treinamento da aplicação, além de não contar com um número suficiente de amostras para um bom aprendizado, pode ser muito contaminado com as características das mensagens, naturalmente curtas e ruidosas. Assim, a combinação de um modelo genérico, treinado com uma grande quantidade de dados, com outro modelo específico, treinado com variações expandidas dos textos da aplicação, pode gerar um modelo mais robusto e apropriado para aplicações de classificação *online* de textos curtos e ruidosos.

É importante notar que em todos os experimentos, as mensagens expandidas foram usadas apenas para treinar os modelos de *embeddings*. No treinamento e teste dos métodos de classificação, eles tiveram acesso apenas as mensagens originais. A premissa é que se as expansões semânticas de fato melhorarem a qualidade das representações vetoriais, o desempenho da classificação *online* será melhor.

5. Resultados e discussões

A Tabela 3 apresenta os resultados obtidos em cada cenário experimental descrito na Seção 4.2. Os valores são apresentados usando tons de cinza na cor de fundo das células, onde células mais escuras representam melhores valores de macro F-medida.

Os resultados obtidos indicam que, tanto nos experimentos com a técnica Word2Vec quanto nos experimentos com a FastText, a expansão semântica de dados ajudou a melhorar os vetores de representação das mensagens e, consequentemente, contribuiu para melhorar o desempenho da classificação *online*. Em sete das nove bases de dados, o melhor resultado nos experimentos com a Word2Vec foi obtido quando foram usados dados expandidos no treinamento do modelo de *embeddings*. Nos experimentos com a FastText, em oito bases de dados, o resultado também foi superior quando os dados expandidos foram usados para treinar o modelo de representação.

Por outro lado, os resultados obtidos no segundo cenário (T7 + aplicação), em geral, foram inferiores aos resultados obtidos no primeiro cenário (T7). Portanto, é possível afirmar que os vetores gerados pelo modelo de representação, treinado apenas com as amostras originais da aplicação, são de baixa qualidade. O baixo desempenho obtido no segundo cenário ajuda a confirmar que a expansão de dados proposta neste trabalho é benéfica para a classificação *online* de textos curtos e ruidosos, pois indica que os resultados apresentados no terceiro cenário (T7 + aplicação expandida) não foram consequência unicamente da concatenação dos vetores da aplicação com os vetores da T7. O fato que mais contribuiu para a hegemonia dos resultados do terceiro cenário foi a expansão das amostras da aplicação. A geração de novas amostras por meio da técnica de expansão semântica ajudou o modelo de representação distribuída a ser mais genérico e robusto aos problemas característicos dos textos das redes sociais, aumentando o número de informações e diminuindo o impacto negativo causado por ruídos, como gírias e abreviações.

Para facilitar a comparação dos resultados e verificar se os desempenhos obtidos com os dados expandidos foram significativamente melhores que os demais, foi realizada uma análise estatística para cada uma das duas técnicas de representação distribuída

Tabela 3. Macro F-medida obtida por cada método de classificação *online* em cada um dos cenários experimentais.

(a) Word2Vec.

Método	T7	T7 + aplicação	T7 + aplicação expandida
Archeage			
NB	0,702	0,728	0,698
PA	0,762	0,784	0,790
Perc.	0,747	0,760	**0,798**
SGD	0,778	0,769	0,784
HCR			
NB	0,627	0,604	0,641
PA	**0,687**	0,672	0,680
Perc.	0,678	0,685	0,684
SGD	0,669	0,671	0,680
Hobbit			
NB	0,810	0,702	0,824
PA	0,850	0,861	**0,875**
Perc.	0,834	0,843	0,854
SGD	0,802	0,842	0,851
IPhone6			
NB	0,646	0,599	0,644
PA	0,705	0,686	0,685
Perc.	0,720	0,692	**0,723**
SGD	0,691	0,718	0,688
OMD			
NB	0,641	0,620	0,652
PA	0,668	0,664	0,698
Perc.	0,681	0,702	**0,708**
SGD	0,674	0,697	0,705
Sanders			
NB	0,664	0,629	0,654
PA	0,757	0,745	**0,786**
Perc.	0,761	0,736	0,765
SGD	0,751	0,752	0,778
SS-Tweet			
NB	0,733	0,666	0,715
PA	0,723	0,738	0,728
Perc.	0,735	0,741	**0,747**
SGD	0,739	0,739	0,743
STS-Test			
NB	0,834	0,562	0,842
PA	0,841	0,822	0,841
Perc.	0,817	0,809	0,833
SGD	**0,849**	0,822	0,825
UMICH			
NB	0,712	0,622	0,714
PA	0,893	0,883	**0,915**
Perc.	0,907	0,873	0,900
SGD	0,884	0,897	0,906

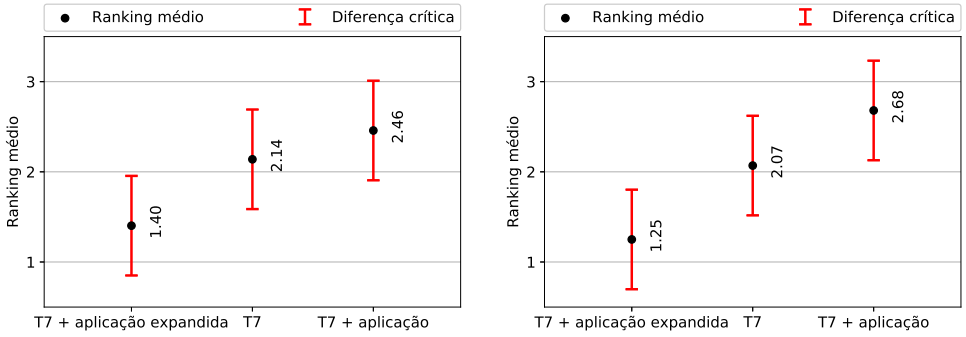
(b) FastText.

Método	T7	T7 + aplicação	T7 + aplicação expandida
Archeage			
NB	0,701	0,589	0,723
PA	0,769	0,791	**0,818**
Perc.	0,784	0,773	0,806
SGD	0,782	0,775	0,812
HCR			
NB	0,616	0,547	0,612
PA	0,648	0,637	**0,688**
Perc.	0,652	0,658	0,669
SGD	0,668	0,663	0,672
Hobbit			
NB	0,770	0,745	0,851
PA	0,850	0,873	**0,907**
Perc.	0,822	0,809	0,850
SGD	0,793	0,818	0,878
IPhone6			
NB	0,695	0,711	0,729
PA	0,707	0,733	0,728
Perc.	0,669	0,727	0,745
SGD	0,715	0,715	**0,747**
OMD			
NB	0,647	0,623	0,662
PA	0,695	0,698	**0,891**
Perc.	0,707	0,699	0,858
SGD	0,720	0,708	0,857
Sanders			
NB	0,676	0,631	0,695
PA	0,781	0,772	**0,906**
Perc.	0,792	0,780	0,889
SGD	0,783	0,759	0,889
SS-Tweet			
NB	0,719	0,675	0,704
PA	0,757	0,725	0,749
Perc.	**0,760**	0,726	0,756
SGD	**0,760**	0,729	0,753
STS-Test			
NB	0,817	0,829	0,825
PA	0,858	0,846	**0,877**
Perc.	0,857	0,834	0,861
SGD	0,874	0,849	0,861
UMICH			
NB	0,829	0,559	0,725
PA	0,923	0,921	**0,977**
Perc.	0,915	0,881	0,965
SGD	0,911	0,884	0,968

(Word2Vec e FastText) usando o teste não paramétrico de Friedman, seguindo cuidadosamente a metodologia descrita em [Demšar 2006]. O teste de Friedman usa o *ranking* médio obtido para cada técnica avaliada para verificar se a hipótese nula, que afirma que os resultados obtidos são estatisticamente equivalentes, pode ser descartada.

Para um intervalo de confiança $\alpha = 0{,}05$ e considerando os *rankings* médios apresentados na Figura 2, o teste de Friedman indicou que a hipótese nula foi rejeitada tanto para os resultados obtidos nos experimentos com a técnica Word2Vec, quanto para os resultados obtidos com a FastText. Diante disso, para verificar se o desempenho obtido com os dados expandidos foi significativamente superior aos demais, foi feita uma análise estatística usando o teste *post-hoc* de Nemenyi [Demšar 2006] (Figura 2). Se a diferença entre os *rankings* for maior do que uma diferença crítica, então conclui-se que há diferença estatística entre os resultados.

Para um intervalo de confiança $\alpha = 0{,}05$, a diferença crítica foi igual a $1{,}69$, tanto nos experimentos com a Word2Vec, como nos experimentos com a FastText. Com base nos *rankings* médios e no valor da diferença crítica, para ambas as análises mostradas na Figura 2, há evidência estatística de que o desempenho obtido com os dados expandidos



(a) Experimentos com a técnica Word2Vec. (b) Experimentos com a técnica FastText.

Figura 2. *Rankings* médios usados nas análises estatísticas e diferenças críticas calculadas usando o teste de Nemenyi.

foi superior ao desempenho obtido apenas com os vetores gerados pelo modelo da T7 e, também, com os vetores gerados pelo modelo da T7 concatenados com os vetores gerados pelo modelo treinado apenas com os dados originais da aplicação.

6. Conclusões e trabalhos futuros

A classificação de mensagens publicadas em redes sociais, como o Twitter, é um problema desafiador, pois elas contêm características que dificultam a geração de um bom modelo de representação de textos: elas costumam ser curtas e repletas de ruídos, como gírias e expressões idiomáticas. Este trabalho propôs uma técnica de expansão semântica de dados para melhorar os modelos de representação distribuída de texto aplicados nesse tipo de problema. Essa técnica usa normalização léxica, indexação semântica e desambiguação para gerar novas mensagens a partir de uma mensagem original e, com isso, aumentar o número de informações que podem ser usadas para treinar o modelo de representação distribuída. As novas amostras são formadas pela combinação dos termos originais, que geralmente são ruidosos, com suas variações normalizadas e expandidas. Portanto, o modelo de representação distribuída consegue relacionar o significados dos termos ruidosos com suas variações canônicas.

Foram realizados experimentos em nove bases de dados públicas, reais, bem conhecidas e que abordam temas variados. Para cada base de dados foram realizados experimentos com as técnicas de representação distribuída Word2Vec e FastText e com os métodos de classificação *online* NB, PA, Perceptron e SGD.

Uma análise estatística dos resultados mostrou que a combinação do modelo de representação distribuída treinado no corpo de texto T7 e o modelo treinado com as mensagens da aplicação expandidas melhorou o desempenho da classificação. Portanto, pode-se afirmar que a expansão semântica, além de ter aumentado o número de informações para a geração dos modelos de representação, ajudou a aproximar os termos ruidosos de suas versões canônicas no espaço multidimensional dos termos. Com isso, depois de treinados, os modelos tornaram-se aptos a lidar com as amostras no seu estado original, sem a necessidade de passar pelas etapas de normalização, indexação e desambiguação. Ao aprimorar a representação vetorial das amostras, a expansão dos dados também melhorou, indiretamente, o desempenho final dos métodos de classificação *online*.

Apesar deste trabalho ter mostrado que a expansão semântica melhora a representação textual das amostras e consequentemente o desempenho da classificação *online*, acredita-se que resultados melhores poderiam ser obtidos se as técnicas de representação distribuída também fossem adaptadas de forma incremental. Portanto, em trabalhos futuros, pretende-se adaptar as técnicas de representação distribuída de textos para que elas também possam evoluir seu aprendizado continuamente, o que é um problema em aberto na literatura.

Agradecimentos

Os autores são gratos à Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP; processos #2017/09387-6 e #2018/02146-6) pelo apoio financeiro a este projeto de pesquisa.

Referências

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., e Passonneau, R. (2011). Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media (LSM'11)*, pages 30–38, Portland, Oregon. Association for Computational Linguistics.

Baziotis, C., Pelekis, N., e Doulkeridis, C. (2017). DataStories at SemEval-2017 task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Bojanowski, P., Grave, E., Joulin, A., e Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Das, M. K., Padhy, B., e Mishra, B. K. (2017). Opinion mining and sentiment classification: A review. In *2017 International Conference on Inventive Systems and Control (ICISC)*, pages 1–3.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

Freund, Y. e Schapire, R. E. (1999). Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3):277–296.

Gama, J., Sebastião, R., e Rodrigues, P. P. (2013). On evaluating stream learning algorithms. *Machine Learning*, 90(3):317–346.

Ghannay, S., Favre, B., Estève, Y., e Camelin, N. (2016). Word embedding evaluation and combination. In Chair), N. C. C., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., e Piperidis, S., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Goikoetxea, J., Agirre, E., e Soroa, A. (2016). Single or multiple? Combining word representations independently learned from text and WordNet. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2608–2614. AAAI Press.

Hirschberg, J. e Manning, C. D. (2015). Advances in natural language processing. *Science*, 349(6245):261–266.

Lochter, J., Zanetti, R., Reller, D., e Almeida, T. (2016). Short text opinion detection using ensemble of classifiers and semantic indexing. *Expert Systems with Applications*, 62:243–249.

Lochter, J. V., Pires, P. R., Bossolani, C., Yamakami, A., e Almeida, T. A. (2018). Evaluating the impact of corpora used to train distributed text representation models for noisy and short texts. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Losing, V., Hammer, B., e Wersing, H. (2018). Incremental on-line learning: A review and comparison of state of the art algorithms. *Neurocomputing*, 275:1261–1274.

McCallum, A. e Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In *Proceedings of the 15th AAAI Workshop on Learning for Text Categorization (AAAI'98)*, pages 41–48, Madison, Wisconsin.

Mikolov, T., Kombrink, S., Burget, L., Černockỳ, J., e Khudanpur, S. (2011). Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5528–5531. IEEE.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., e Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS'13)*, pages 3111–3119, Lake Tahoe, Nevada, USA. Curran Associates Inc.

Moro, A., Raganato, A., e Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., e Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Saito, I., Suzuki, J., Nishida, K., Sadamitsu, K., Kobashikawa, S., Masumura, R., Matsumoto, Y., e Tomita, J. (2017). Improving neural text normalization with data augmentation at character- and morphological levels. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 257–262. Asian Federation of Natural Language Processing.

Shamma, D. A., Kennedy, L., e Churchill, E. F. (2009). Tweet the debates: Understanding community annotation of uncollected sources. In *Proceedings of the First SIGMM Workshop on Social Media (WSM'09)*, pages 3–10, Beijing, China. ACM.

Silva, R. M., Alberto, T. C., Almeida, T. A., e Yamakami, A. (2017). Towards filtering undesired short text messages using an online learning approach with semantic indexing. *Expert Systems with Applications*, 83:314–325.

Socher, R. (2015). *Recursive Deep Learning for Natural Language Processing and Computer Vision*. PhD thesis, Stanford University.

Speriosu, M., Sudan, N., Upadhyay, S., e Baldridge, J. (2011). Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First Workshop on Unsupervised Learning in NLP (EMNLP'11)*, pages 53–63, Edinburgh, Scotland. Association for Computational Linguistics.

Thelwall, M., Buckley, K., e Paltoglou, G. (2012). Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology*, 63(1):163–173.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., e Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., e Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Vo, D. e Ock, C. (2015). Learning to classify short text from scientific documents using topic models with various types of knowledge. *Expert Systems with Applications*, 42(3):1684–1698.

Xie, Q., Dai, Z., Hovy, E. H., Luong, M., e Le, Q. V. (2019). Unsupervised data augmentation. *CoRR*, abs/1904.12848.

Yang, J. e Leskovec, J. (2011). Patterns of temporal variation in online media. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 177–186, New York, NY, USA. ACM.

Zhang, T. (2004). Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the 21th International Conference on Machine Learning (ICML'04)*, pages 116–123, Banff, Alberta, Canada. ACM.