Análise de Sentimentos: Avaliando o Desempenho de Pré-Processamento e de Algoritmos de Aprendizagem de Máquina sobre o Dataset TweetSentBR

Laerte dos Santos Cardozo[1], **Larissa Astrogildo de Freitas**[1]

[1]Centro de Desenvolvimento Tecnológico (CDTec)
Universidade Federal de Pelotas (UFPel) – Pelotas, RS – Brasil

{ldscardozo,larissa}@inf.ufpel.edu.br

Abstract. *Sentiment analysis is the field of study responsible for analyzing the opinions contained in applications or services, which are being used a lot by society. In this work a sentiment analysis was performed in the TweetSentBR dataset, in which machine learning techniques were used to identify the sentiments that were contained in the texts. Preprocessing approaches were applied for the algorithms to classify them. The best results showed an F-Measure with a prediction of 82% for binary polarity using SVM.*

Resumo. *A análise de sentimentos é o campo de estudo responsável por analisar as opiniões contidas em aplicações ou serviços, os quais estão sendo muito utilizados pela sociedade. Neste trabalho, foi realizada uma análise de sentimentos no dataset TweetSentBR, na qual foram utilizadas técnicas de aprendizado de máquina para identificação dos sentimentos contidos nos textos. Foram aplicadas abordagens de pré-processamento para os algoritmos os classificarem. Os melhores resultados apresentaram uma Medida F com predição de 82% para a polaridade binária utilizando SVM.*

1. Introdução

Nos dias de hoje, as tecnologias que estão à disposição geram grandes volumes de informações através de seus usuários. De acordo com [Martins et al. 2015], redes sociais, sites de notícias, fóruns de discussão e blogs são exemplos de geradores de informações que agrupam grandes quantidades de dados.

Uma das tarefas de Processamento da Língua Natural (PLN), mais especificamente, de compreensão da linguagem natural, é a Análise de Sentimento (AS). Segundo [Liu 2012], a AS analisa opiniões, sentimentos e emoções dos usuários em relação a produtos, organizações, serviços e outros. Conforme [Benevenuto et al. 2015], a polaridade na AS representa o grau de quanto positivo ou negativo é um texto. Suas principais representações são: binário (positivo e negativo) ou ternário (positivo, negativo e neutro).

No trabalho de [Brum and Nunes 2018], os resultados apresentados com a análise para a polaridade ternária são inferiores aos resultados apresentados na polaridade binária. Dessa forma, a motivação para este trabalho é melhorar os resultados apresentados em [Brum and Nunes 2018] com outros algoritmos de Aprendizado de Máquina (AM) (tais como: SVM [Cortes and Vapnik 1995] e LSTM [Hochreiter and Schmidhuber 1997]) e adicionar outras abordagens de pré-processamento de textos (tais como: filtros, remoção

de *stopwords*, correção ortográfica e *stemming*). O objetivo deste trabalho é igualar ou superar os resultados apresentados pelos autores do TweetSentBR [Brum and Nunes 2018].

Este artigo está estruturado da seguinte forma: a Seção 2 mostra os trabalhos relacionados; a Seção 3 apresenta as técnicas utilizadas para o pré-processamento e os algoritmos de AM; a Seção 4 mostra os resultados obtidos com as técnicas descritas na seção anterior; e a Seção 5 apresenta as conclusões e os trabalhos futuros.

2. Trabalhos Relacionados

Perante o objetivo deste trabalho, pesquisou-se na literatura por trabalhos que utilizaram diferentes abordagens de pré-processamento de textos e que utilizaram algoritmos de AM. Dentre as técnicas de pré-processamento, pode-se citar a remoção de *stopwords* que consiste em remover palavras que são consideradas irrelevantes para a AS (por exemplo: preposições e artigos). O *stemming* reduz as palavras para os seus radicais, removendo os afixos (prefixo e sufixo), permitindo que diversas derivações de uma mesma palavra possam ser combinadas [Coelho 2007].

No artigo de [Brum and Nunes 2018] foi proposta a construção do *dataset* TweetSentBR no qual foi utilizado AM para a classificação dos sentimentos. Nesse mesmo trabalho, os autores utilizaram diferentes técnicas de pré-processamento: os números foram substituídos pela *tag* NUMBER, os nomes de usuários por USERNAME e os *links* por URL. Além disso, foram limitadas as repetições de um caractere em uma mesma palavra para corrigir sua grafia. Os algoritmos de AM utilizados foram: SVM, *Bernoulli* NB, Regressão Logística (LR), *Multilayer Perceptron*, Árvore de Decisão e *Random Forest*. Em [Nascimento 2019], foi proposta a identificação de sentimentos em textos curtos com AM, um dos *datasets* utilizados foi o TweetSentBR. Os algoritmos de AM implementados foram: LR, *Gaussian* NB, KNN, *Multilayer Perceptron*, SGD e SVM. No pré-processamento, foram utilizados filtros e remoção de *stopwords*.

Em [Kumar and Subba 2020], foi proposto um *framework* de AS utilizando o algoritmo SVM para a classificação de sentimentos em textos. Além disso, para o pré-processamento foram utilizados a remoção de *stopwords*, filtros (remoção de *emoticons* e caracteres especiais) e *stemming*. Em [Sakiyama et al. 2020], foi proposta uma abordagem utilizando AM para predizer quais os participantes de um *reality show* seriam eliminados com base nos *tweets*. No pré-processamento, foram preservados os USERNAMES e foram removidos números, porcentagens, URLs e datas.

3. Metodologia

Nesta seção, é apresentada a descrição do trabalho desenvolvido para classificar os sentimentos contidos em textos utilizando como base de dados: o TweetSentBR. A abordagem proposta recebe como entrada um texto e o classifica com o sentimento predito por um dos algoritmos de AM supervisionado (Fig. 1), que será detalhada nas Seções 3.1, 3.2, 3.3 e 3.4.

3.1. Dataset

O TweetSentBR é um *dataset* para AS para o português brasileiro, o qual possui 15.000 textos (6.648 positivos, 3.926 neutros e 4.426 negativos) com diversos assuntos sobre

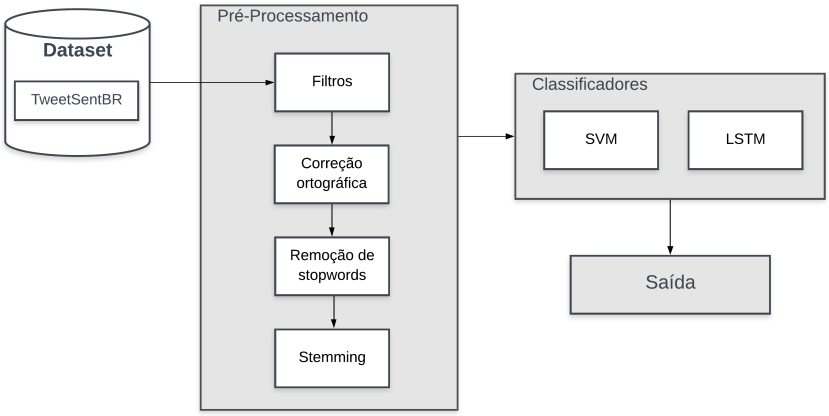

Figura 1. Estrutura da abordagem proposta.

programas brasileiros de televisão. Seus textos foram rotulados com três classes de sentimentos (positivo, negativo e neutro) por sete anotadores [Brum and Nunes 2018]. Para este trabalho, o *dataset* foi mantido desbalanceado para ser realizada uma comparação com os resultados de [Brum and Nunes 2018].

3.2. Pré-processamento

No pré-processamento, foram utilizados filtros, correção ortográfica, remoção de *stopwords* e *stemming*. Tendo como principal objetivo melhorar os textos para a etapa de classificação de sentimentos. Para exemplificar, foi retirada a seguinte frase do *dataset*: "falta de educação da pessoa no celular no programa". Após realizar o pré-processamento, a frase resultante foi: "falt educ pesso celul program".

Os filtros foram implementados para diminuir os ruídos que possam estar contidos nos textos. Para padronizar a escrita das palavras nos textos, as letras foram convertidas para minúsculas. Palavras que são muito utilizadas no Twitter, como *usernames*, *hashtag*, *emojis*, *e-mail* e *links* de *sites* foram substituídas, respectivamente, por: USERNAME, HASHTAG, EMOJIS, EMAIL e SITE. Os números foram substituídos por NUMBER. Acentos ortográficos e pontuações foram removidos para padronizar os textos. Também foram removidos os espaços desnecessários, causados tanto por digitação quanto pelas substituições ou remoções dos filtros citados anteriormente.

A correção ortográfica foi utilizada para corrigir a grafia das palavras. De acordo com [Britto and Pacífico 2019], erros ortográficos e abreviaturas são comumente encontrados em *corpus* oriundos da Web. O corretor foi implementado com a biblioteca *SymSpellpy*[1], sendo utilizado um dicionário com 50 mil palavras em português brasileiro.

As *stopwords* foram removidas dos textos, exceto as que possuem sentido negativo, como "não", "nenhum", "nada", "jamais", "nunca" e "nem". Essa etapa foi implementada utilizando as listas de *stopwords* das bibliotecas do NLTK[2] e do *spaCy*[3]. O *stemming* foi implementado com o RSLP (*Removedor de Sufixos da Língua Portuguesa*) e com o *Snowball*. O RSLP foi desenvolvido por [Orengo and Huyck 2001] para ser um

[1] `https://pypi.org/project/symspellpy/`
[2] `https://www.nltk.org/`
[3] `https://spacy.io/`

algoritmo totalmente projetado para o português. Para o *Snowball* [Porter 2001] foi utilizado o idioma português, sendo que esse suporta diversos idiomas para o *stemming*.

3.3. Classificadores

Após o processo de pré-processamento, os classificadores são utilizados para predizerem os sentimentos contidos nos textos. Para esta etapa foram implementados os algoritmos: SVM e LSTM.

De acordo com [Russell and Norvig 2013], o SVM é uma técnica utilizada para o reconhecimento de padrões sobre o espaço vetorial. Esta técnica é capaz de encontrar um plano de separação entre informações de diferentes classes, conhecido como hiperplano. Para este trabalho, foi utilizado o *LinearSVC* com o mesmo parâmetro de [Brum and Nunes 2018] e o TF-IDF foi implementado com os mesmos parâmetros utilizados em [Kumar and Subba 2020].

Segundo [Hochreiter and Schmidhuber 1997], o LSTM foi desenvolvido para resolver o problema de dependência a longo prazo em dados sequenciais. Neste trabalho, foram utilizados no LSTM os mesmos parâmetros de [Britto and Pacífico 2019].

3.4. Saída

A saída é o sentimento recebido das predições realizadas pelos algoritmos de AM supervisionado para a classificação dos textos.

4. Resultados

Nesta seção, é apresentada uma comparação dos resultados obtidos no trabalho de [Brum and Nunes 2018] com a abordagem proposta na Seção 3. O *dataset* TweetSentBR utilizado nos experimentos foi mantido desbalanceado para a comparação dos resultados.

Nos algoritmos (SVM e LSTM), foram realizadas diferentes combinações para o pré-processamento dos textos, sendo que algumas obtiveram os mesmos resultados. Para a polaridade ternária, o melhor resultado no LSTM foi utilizando o *stemming* do *Snowball* e a remoção das *stopwords* do *spaCy*. No SVM foram: remoção das *stopwords* do NLTK e *stemming* do RSLP; remoção das *stopwords* do NLTK e *stemming* do *Snowball*; remoção das *stopwords* do *spaCy* e *stemming* do *Snowball*. Para a polaridade binária, os melhores resultados com pré-processamento no LSTM foram: remoção das *stopwords* do NLTK e *stemming* do RSLP; remoção das *stopwords* do *spaCy* e *stemming* do *Snowball*; remoção das *stopwords* do NLTK juntamente com as *stopwords* do *spaCy* e *stemming* do RSLP; remoção das *stopwords* do NLTK juntamente com as *stopwords* do *spaCy* e *stemming Snowball*. No SVM, foram: remoção das *stopwords* do NLTK e *stemming* do RSLP; remoção das *stopwords* do NLTK e *stemming* do Snowball.

Além disso, foi utilizada validação cruzada com 10 *folds* nos experimentos para o treino e teste dos textos nos algoritmos. Como o dataset está desbalanceado, os textos foram validados com a métrica de Medida F.

Na Tabela 1, são apresentados os resultados com a polaridade ternária. O LR foi o método do trabalho de [Brum and Nunes 2018] que obteve o melhor resultado para esse tipo de polaridade. Nota-se que em todos os métodos, a polaridade com o sentimento

Tabela 1. Classificação com polaridade ternária.

Autor	Pré-processado	Método	F-Pos	F-Neu	F-Neg	Medida F
[Brum and Nunes 2018]	Sim	LR	**76.6%**	**51.7%**	**66.3%**	**64.87%**
Proposta	Sim	LSTM	70.0%	35.0%	59.0%	55.00%
Proposta	Sim	SVM	75.0%	46.0%	65.0%	62.00%
Proposta	Não	LSTM	73.0%	42.0%	59.0%	58.00%
Proposta	Não	SVM	75.0%	44.0%	65.0%	61.00%

neutro obteve os menores resultados, descrevendo a dificuldade para os algoritmos reconhecerem os textos que apresentem esse tipo de sentimento.

Na Tabela 2, são apresentados os resultados com a polaridade binária. Verifica-se que com a remoção do sentimento neutro houve um aumento nos resultados para as outras polaridades. Isso se deve pela diminuição da complexidade no problema de identificação de sentimentos em textos.

Tabela 2. Classificação com polaridade binária.

Autor	Pré-processado	Método	F-Pos	F-Neg	Medida F
[Brum and Nunes 2018]	Sim	LR	84.78%	75.99%	80.38%
Proposta	Sim	LSTM	82.00%	72.00%	77.00%
Proposta	Sim	SVM	**86.00%**	**77.00%**	**82.00%**
Proposta	Não	LSTM	83.00%	74.00%	79.00%
Proposta	Não	SVM	85.00%	75.00%	80.00%

Ainda, nas Tabelas 1 e 2, verifica-se que no SVM ao não aplicar o pré-processamento nos textos, na polaridade ternária houve uma diminuição no resultado da Medida F. Na polaridade binária houve um aumento de 2% no resultado ao utilizar o pré-processamento. Para o LSTM nota-se que ao não utilizar o texto pré-processado houve um aumento nos resultados, tanto da polaridade binária quanto da polaridade ternária.

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou uma AS no *dataset* TweetSentBR utilizando AM para a classificação dos sentimentos em textos. Diferentes técnicas de pré-processamento foram utilizadas com o objetivo de ajudar os algoritmos de AM na identificação dos sentimentos. Sendo implementado com filtros, correção ortográfica, remoção de *stopwords* e *stemming*. Neste trabalho, um dos resultados obtidos proporcionou uma Medida F de predição com 82% para a polaridade binária. Esse resultado foi superior aos obtidos por [Brum and Nunes 2018] para a polaridade binária. Contudo, para a polaridade ternária, os resultados obtidos foram inferiores aos obtidos por [Brum and Nunes 2018], 62% e 64,87%, respectivamente. O código fonte utilizado neste trabalho está disponível *online*[4].

[4]`https://github.com/laertecardozo/Analise-de-Sentimentos-Portugues`

Como trabalhos futuros, pretende-se investigar o impacto do pré-processamento nos resultados e utilizar outras técnicas de AM ou *Deep Learning*, como LR, CNN e *Transformers* (BERT).

Referências

[Benevenuto et al. 2015] Benevenuto, F., Ribeiro, F., and Araújo, M. (2015). Métodos para análise de sentimentos em mídias sociais. In *Proceedings of the Brazilian Symposium on Multimedia and the Web (Webmedia)*, Manaus, Brasil. Webmedia.

[Britto and Pacífico 2019] Britto, L. F. S. and Pacífico, L. D. S. (2019). Sentiment analysis for mobile app reviews in brazilian portuguese. In *2019 Encontro Nacional de Inteligência Artificial e ComputaÇão (ENIAC 2019)*, volume 1, pages 1–12, Salvador. SBC.

[Brum and Nunes 2018] Brum, H. B. and Nunes, M. G. V. (2018). Building a sentiment corpus of tweets in brazilian portuguese. In *Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japão. European Language Resources Association (ELRA).

[Coelho 2007] Coelho, A. R. (2007). Stemming para a língua portuguesa: estudo, análise e melhoria do algoritmo rslp. Monografia (Graduação em Ciência da Computação), UFRGS (Universidade Federal do Rio Grande do Sul), Porto Alegre, Brasil.

[Cortes and Vapnik 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

[Hochreiter and Schmidhuber 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

[Kumar and Subba 2020] Kumar, V. and Subba, B. (2020). A tfidfvectorizer and svm based sentiment analysis framework for text data corpus. In *2020 National Conference on Communications (NCC)*, pages 1–6.

[Liu 2012] Liu, B. (2012). *Sentiment analysis and opinion mining*, volume 5 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers, USA.

[Martins et al. 2015] Martins, R. F., Pereira, A., and Benevenuto, F. (2015). An approach to sentiment analysis of web applications in portuguese. In *Proceedings of the 21st Brazilian Symposium on Multimedia and the Web*, pages 105–112, New York, NY, USA. ACM, Association for Computing Machinery.

[Nascimento 2019] Nascimento, P. A. (2019). Aplicando ensemble para classificação de textos curtos em português do brasil. Dissertação (Mestrado em Ciência da Computação), Universidade Federal de Pernambuco (UFPE), Recife, Brasil.

[Orengo and Huyck 2001] Orengo, V. and Huyck, C. (2001). A stemming algorithmm for the portuguese language. In *String Processing and Information Retrieval, International Symposium on*, page 0186, Los Alamitos, CA, USA. IEEE Computer Society.

[Porter 2001] Porter, M. F. (2001). Snowball: A language for stemming algorithms. Disponível em: `http://snowball.tartarus.org/texts/introduction.html`. Acesso em: 17 dez. 2020.

[Russell and Norvig 2013] Russell, S. and Norvig, P. (2013). *Inteligência Artificial*. Elsevier Editora, Rio de Janeiro, 3 edition.

[Sakiyama et al. 2020] Sakiyama, K., de Souza Rodrigues, L., and Matsubara, E. T. (2020). Can twitter data estimate reality show outcomes? In *Intelligent Systems*, pages 466–482, Cham. Springer International Publishing.