Análise de modelos de classificação de texto para o problema do sexismo

Lisle Faray de Paiva[1], **Alana Cristina de Carvalho Araújo**[1], **Giovanna Pavanni Esteve**[1], **Anselmo Cardoso de Paiva**[1]

[1]Universidade Federal do Maranhão (UFMA)
Av. dos Portugueses, 1966 - Vila Bacanga, São Luís - MA, 65080-805

{faray.lisle, alana.cca, giovanna.esteve}@discente.ufma.br, paiva@nca.ufma.br

Abstract. *Social media has been a space where users feel free to report their opinions and feelings, leading to an abundance of abusive and hateful comments that require a way to filter them so that their moderation can occur. In order to classify sexist comments on the Twitter platform, three different classifiers were built: Logistic Regression, a Multi-Layer Perceptron(MLP), and a Convolutional Neural Network(CNN), and thus tested so that it was possible to identify which would obtain better results for this specific problem. The best result was achieved by the CNN alchitecture that managed to reach an F1-score of 92.42%, an accuracy of 93.19%, and a precision of 96.77%. Thus, it is possible to conclude that the CNN is a classifier with good results when applied in the text classification task.*

Resumo. *Redes sociais tem sido um espaço onde usuários se sentem livres para relatar suas opiniões e sentimentos, o que pode levar a uma abundância de comentários abusivos e cheios de ódio que requerem uma forma de filtrá-los para que possa ocorrer sua moderação. Com o objetivo de classificar comentários sexistas na plataforma do Twitter, foram construídos três classificadores diferentes: Regressão Logística, Multi-Layer Perceptron(MLP) e uma Rede Neural Convolucional(CNN), e assim testados, para que fosse possível identificar qual iria obter melhores resultados para este problema específico. O melhor resultado foi alcançado pela arquitetura CNN que conseguiu alcançar um F1-score de 92,42%, uma acurácia de 93,19% e precisão de 96,77%. Assim, é possível concluir, que a CNN é um classificador com resultados muito bons quando aplicado na tarefa de classificação de textos.*

1. Introdução

Atualmente, a Internet tem sido um dos meios mais populares para a disseminação de preconceitos e discriminações podendo aparecer na forma de comentários, *tweets* e mensagens, como no exemplo mostrado na Figura 1, onde é apresentada a comparação entre um *tweet* normal e um *tweet* com características de discriminação. As redes sociais ainda buscam formas de tentar filtrar esse tipo de mensagem e ideia, para conseguir diminuir essa dispersão negativa entre os usuários, como por exemplo, no caso do *Twitter*, que anunciou, na própria plataforma, que irá oferecer recompensas em dinheiro para usuários e pesquisadores que descobrirem possíveis vieses sexistas ou racistas em um dos algoritmos da rede [G1 2021]. Existem diversos tipos de intolerância e repúdio que são propagados todos os dias, como, racismo, homofobia, gordo-fobia, etc.

O sexismo trata-se de atitudes discriminatórias, sejam ações ou discurso que agridam, ofendam ou diminuam pessoas de um certo gênero [Braga et al. 2021]. E, na Internet, a forma mais comumente apresentada do sexismo é voltada contra a mulher. Neste trabalho, será aplicada a detecção do sexismo.

(a) (b)

Figura 1. (a) Tweet normal; (b) Tweet sexista

Uma das formas de conseguir identificar a publicação dessas ideias é através da utilização de *Deep Learning* para a classificação de mensagens. De acordo com [El Naqa and Murphy 2015], Aprendizado de Máquina é uma ramificação de algoritmos que são desenvolvidos para imitar a inteligência humana aprendendo a partir do ambiente ao seu redor. De acordo com [Ikonomakis et al. 2005], quando a mesma é utilizada na classificação de texto, é necessário que ela seja aplicada em documentos ou termos sob uma categoria pré-definida.

A partir desses conceitos, é possível entender o que será trabalhado ao longo desse artigo que busca propor uma forma de classificar *tweets* entre duas categorias, sexistas ou não sexistas. Esse artigo é divido em 5 seções, Introdução, onde são apresentados os conceitos iniciais do que será abordado, trabalhos relacionados, em que são apresentados artigos que trabalham temas semelhantes ao deste trabalho. Na terceira seção, a metodologia, onde são apresentadas as etapas realizadas no desenvolvimento do trabalho para realizar a classificação das mensagens em relação ao viés de sexismos, seguida da seção dos resultados, em que são apresentados os rendimentos das métricas encontrados no treinamento. E, finalmente, é apresentada a conclusão do trabalho onde que serão mostrados os conhecimentos adquiridos e os trabalhos futuros.

2. Trabalhos Relacionados

Em [Fortuna 2017], os autores construíram um *dataset* de discurso de ódio selecionando sites, incluindo o twitter, que incentivassem a interação de usuário. Foram coletados diferentes tipos de informações com o objetivo de obter uma quantidade representativa de comentários ao invés de um tipo específico deles.

Os autores treinaram um modelo *deep cross-lingual Long Short-Term Memory* (LSTM) com o dataset pré-processado e vetorizado. As principais abordagens adotadas consideraram o treinamento de *embeddings* através de vetores de índices de palavras, vetores TF-IDF e vetores *n-Grams*. [Fortuna 2017] também testou o uso de uma rede neural do tipo *Feed-Foward* para a classificação de texto para o problema do discurso de ódio obtendo 77,3% de acurácia e 80,4% de precisão.

Das contribuições na língua portuguesa, [Malta and Kuroiva 2019] utiliza de técnicas de processamento de linguagem natural e aprendizado de máquina em um sistema de moderação de comentários automático. Os autores utilizaram um modelo Cross Industry Standard Process for Data Mining (CRISP-DM) treinado com o dataset Off-ComBR [de Pelle and Moreira 2017].

A metodologia desse trabalho se baseou na aplicação de diversas fases, na forma de um ciclo de vida da mineração de dados. As fases aplicadas no modelo foram a de entendimento de negócios, entendimento de dados, preparação de dados, modelando, avaliação e a implantação. Para a avaliação dos resultados do segundo ciclo de classificação, foram utilizadas as métricas de precisão, sensibilidade e acurácia. A performance mais equilibrada foi obtida pelo modelo que representava a junção do Random Forest com o MLP e o SVC obtendo 48% de precisão, 45% de recall e 77% de acurácia.

Já o [Bispo 2018] aborda uma metodologia que utiliza um modelo *Deep Cross-Lingual Long Short-Term Memory*(LSTM) para fazer uma classificação hierárquica na detecção de discurso de ódio. Para isso, conduziu-se uma experiência comparando duas abordagens diferentes: *unimodel*, onde o modelo considera apenas a classe discurso de ódio; e *multimodel*, onde o modelo considera diversas classes organizadas hierarquicamente.

Buscando detectar qual abordagem seria mais promissora para o trabalho, [Bispo 2018] aplicou diversas formas de pré-processamento e de vetorização das bases de dados representados por 24 cenários diferentes aplicados nos experimentos. Dentre os cenários apresentados, o que possuiu resultados mais satisfatórios para o presente trabalho foi cenário 23 tratando-se de uma LSTM treinada com NAACL_SRW_2016_cleaned_pt lematizado, seu próprio vocabulário e testada com discursos_votados lematizado com 70,3% de precisão.

Entre todos os trabalhos relacionados, nenhum aborda o problema específico do sexismo. Todos eles lidam com o problema do discurso de ódio, um problema mais abrangente que engloba não só o sexismo, como também o racismo, homofobia entre outros. Portanto, este trabalho visa lidar com a classificação de textos sexistas.

3. Metodologia

Este trabalho apresenta uma metodologia para classificação de textos na língua inglesa de conotação sexista como ilustrado na Figura 2 visando analisar três métodos de classificação de texto. Primeiramente é montada a base de dados a partir do conjunto de dados público *Classified Tweets* [Albright 2021]. Então segue-se para a vetorização dos textos, estes serão utilizados como vetores de características para o classificador linear e a rede neural *feed-foward*. Para a classificação com a rede neural convolucional, em vez de usar os vetores de caracteristicas é utilizado *word embeddings* como entrada para o modelo. Após o pré-processamento os vetores e *word embeddings* seguem para seus respectivos modelos para serem classificados e avaliados.

3.1. Aquisição de dados e pré-processamento

A base de dados utilizada nessa metodologia é composta de dados obtidos de uma base pública de *tweets* na língua inglesa classificados entre *tweets* sexistas, racistas ou nenhum dos dois chamada *Classified Tweets* [Albright 2021]. Da base pública foram selecionados

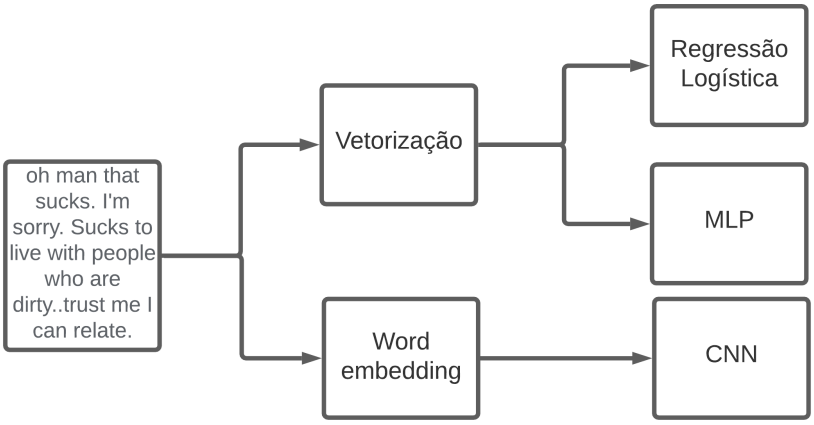

Figura 2. Fluxograma da metodologia proposta.

1733 *tweets* sexistas e 1733 *tweets* classificados como não racistas e não sexistas. Tendo os dados selecionados e separado em suas devidas categorias, segue-se para o pré processamento do texto que será explicado mais a fundo a seguir. Na tabela 1 pode-se observar um exemplo do contéudo presente nos *tweets* utilizados.

Classe	Tweets
sexista	u stll up drinking bitch?
não-sexista	damn! you packin heat!

Tabela 1. Exemplo de tweets do dataset.

3.1.1. Vetorização do texto

A vetorização dos textos permite que maquinas entendam seu conteúdo ao converte-los em representações numéricas significativas [Fredigo Hack et al. 2013]. A forma mais comum de vetorização de textos é feita transformando-os em uma matriz de frequência, onde cada documento é representado por um vetor de termos e cada termo possui um valor associado que indica o grau de importância desse no documento.

Para este trabalho uma versão da base de dados é tokenizada e normalizada utilizando a biblioteca apresentada em [Pedregosa et al. 2011], os textos normalizados são então transformados em matrizes esparsas. As matrizes esparsas vão então ser utilizados como entrada para os modelos de classificação Regressão Logística e a rede neural *Feed-Foward*.

3.1.2. Word embeddings

Word embeddings são um tipo de representação vetorial de texto que permite que palavras com significados similares tenham representações similares. Essa forma de representação de palavras pode ser considerada uma importante avanço do aprendizado profundo em problemas desafiadores de processamento de linguagem natural

[Goldberg and Hirst 2017].

Para este trabalho uma versão da base de dados é tokenizada e convertida para sequências de caracteres, as sequências são então transformadas em *arrays numpy 2D* através da técnica de *padding* de sequências, esses *arrays* são utilizados como *word embeddings* como entradas para o treinamento e teste da rede neural convolucional implementada.

3.2. Classificação do texto

Após o pré-processamento, segue-se para a classificação dos textos. São realizados três experimentos com três classificadores diferentes: o classificador Regressão Logística, uma rede neural do tipo *Feed-foward* e uma Rede Neural Convolucional que serão detalhados mais a fundo a seguir.

3.2.1. Regressão Logística

Pode-se entender regressão logística como o análogo de regressão linear para problemas de classificação. De acordo com [Figueira 2006], ela permite a utilização de um modelo para encontrar a probabilidade de um evento especifico. Ela também é caracterizada pelo fato de não fazer suposições a partir da forma funcional das suas variáveis independentes, o que significa que ela pode ser utilizada de maneira mais generalizada.

Enquanto que na regressão linear, usa-se o método dos mínimos quadrados, a regressão logística utiliza o método da máxima verossimilhança e a variável dependente é categórica. Na regressão logística, a probabilidade de ocorrências de um evento pode ser estimada diretamente.

Utilizando o texto vetorizado como entrada, utiliza-se o classificador Regressão Logística para a classificação. Foi utilizada a implementação disponibilizada na biblioteca *scikit-learning* [Pedregosa et al. 2011] utilizando a parametrização *default*.

3.2.2. Rede Neural Feed-foward

Segundo [Law 2000], uma Rede Neural Feed-Foward pode ser caracterizada pela sua habilidade em encontrar padrões ao longo do treinamento. Ela é formada por três camadas que somente se conectam em uma direção, onde vai da camada de entrada, passando pela camada intermediária e finalizando na camada de saída da rede.

Para este trabalho é implementada uma *Multi Layer Perceptron*(MLP) tendo como sua última camada uma ativação Sigmoid descrita na Equação 1. A MLP utiliza a base de dados vetorizada como entrada para classificação.

$$S(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

3.2.3. Rede Neural Convolucional

De acordo com [Vargas et al. 2016], uma Rede Neural Convolucional (CNN) pode ser definida como uma rede dividida em diversas partes, onde cada parte possui uma função diferente. Ela é muito utilizada em projetos de classificação, detecção e reconhecimento em imagens e vídeos, justamente pelo fato de conseguir aplicar filtros em dados visuais.

Para este trabalho é implementada uma CNN com uma camada convolucional, onde sua última camada é uma ativação Sigmoid conforme a Equação 1. Como entrada a CNN utiliza os *word embeddings* como vetores de características para classificação.

4. Resultados e Discussão

Para avaliar a metodologia proposta foram usadas as seguintes métricas de avaliação: acurácia, sensibilidade, especificidade, precisão e *F1-score*, descritas nas Equações 2, 3, 4, 5 e 6, respectivamente. Assim, busca-se uma metodologia que tenha a capacidade de classificar corretamente as duas classes consistentemente.

$$ACU = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$SEN = \frac{TP}{TP + FN} \tag{3}$$

$$ESP = \frac{TN}{TN + FP} \tag{4}$$

$$PRE = \frac{TP}{TP + FP} \tag{5}$$

$$F1 = \frac{2 \cdot PRE \cdot SEN}{PRE + SEN} \tag{6}$$

O estado da arte [Lin et al. 2021] para classificação de texto implementa um modelo que compreende de um codificador de documento e um codificador de contexto, que usa a camada *Graph Convolutional Networks* (GCN) e *Bidirectional Encoder Representations from Transformers* (BERT), que é um modelo pré-treinado de dados textuais treinado em cinco bases de dados obtendo acurácias médias para o modelo BERT GCN variando entre 72,8% a 98,2% .

Analisando as acurácias obtidas nos três experimentos observaods na Tabela 2, pode-se observar que todas as métricas se encontram acima de 90%. Fazendo um comparativo com o estado da arte, a acurácia dos modelos aqui implementados chegam próximas das dos modelos descritos por Lin et al. mostrando que os resultados obtidos foram satisfatórios.

Classificador	Sensibilidade	Especificidade	Precisão	Acurácia	F1-score
Regressão Logística	87,71%	96,52%	95,71%	92,38%	91,53%
MLP	86,97%	95,86%	94,90%	91,69%	90,76%
CNN	88,45%	97,39%	96,77%	93,19%	92,42%

Tabela 2. Resultados obtidos a partir dos classificadores.

Para o problema da classificação de sexismo, ao traçar um comparativo entre a sensibilidade e a especificidade pode-se observar que, para todos os modelos, a especificidade é significativamente superior à sensibilidade, o que indica que os modelos melhor identificaram *tweets* que não se enquadram como sexistas do que o oposto, isso pode estar relacionado com os modelos em si ou com a base de dados e, por isso, deve-se investigar o uso de outras bases de dados.

5. Conclusão

O objetivo principal desse trabalho foi fazer a classificação de textos sexistas da rede social chamada Twitter. Foram realizados três experimentos utilizando três classificadores Regressão Logística, MLP e CNN, para que fosse possível encontrar aquele que resultaria em um melhor resultado. Apesar de todos os três algoritmos terem apresentados resultados próximos e satisfatório, a CNN obteve os melhores valores em todas as métricas, e assim foi definido como a melhor arquitetura analisada para o nosso problema.

Em trabalhos futuros, pretende-se resolver alguns problemas observados ao longo da construção dos algoritmos, como por exemplo, o *overfitting* da arquitetura CNN explorando outras bases de dados. Pretende-se também analisar e implementar outros métodos de classificação para o problema do sexismo e utilizar os *word embeddings* em outros classificadores.

Referências

Albright, M. (2021). Classified tweets. https://www.kaggle.com/munkialbright/classified-tweets.

Bispo, T. D. (2018). Arquitetura lstm para classificação de discursos de ódio cross-lingual inglês-ptbr.

Braga, M. L. P., Nakamura, F. G., and Nakamura, E. F. (2021). Criação e caracterização de um corpus de discurso sexistas em português. *iSys-Brazilian Journal of Information Systems*, 14(2):79–95.

de Pelle, R. P. and Moreira, V. P. (2017). Offensive comments in the brazilian web: a dataset and baseline results.

El Naqa, I. and Murphy, M. J. (2015). What is machine learning? In *machine learning in radiation oncology*, pages 3–11. Springer.

Figueira, C. V. (2006). Modelos de regressão logística.

Fortuna, P. C. T. (2017). Automatic detection of hate speech in text: an overview of the topic and dataset annotation with hierarchical classes.

Fredigo Hack, A., Felipe Nunes, L., Hoffmann Silva, M., and Thalison F de Lima, T. (2013). *Text Mining*. UFSC, Universidade Federal de Santa Catarina.

G1 (2021). Twitter oferece recompensas a quem corrigir vieses raciais e sexistas em seus algoritmos de corte de imagens. https://g1.globo.com/economia/tecnologia/noticia/2021/08/02/twitter-oferece-recompensas-a-quem-corrigir-vieses-raciais-e-sexistas-em-seus-algoritmos-de-corte-de-imagens.ghtml.

Goldberg, Y. and Hirst, G. (2017). Neural network methods in natural language processing. morgan & claypool publishers(2017). *9781627052986 (zitiert auf Seite 69)*.

Ikonomakis, M., Kotsiantis, S., and Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS transactions on computers*, 4(8):966–974.

Law, R. (2000). Back-propagation learning in improving the accuracy of neural network-based tourism demand forecasting. *Tourism Management*, 21(4):331–340.

Lin, Y., Meng, Y., Sun, X., Han, Q., Kuang, K., Li, J., and Wu, F. (2021). Bertgcn: Transductive text classification by combining gcn and bert. *arXiv preprint arXiv:2105.05727*.

Malta, L. H. A. and Kuroiva, M. A. R. L. (2019). Aprendizado de máquina e processamento de linguagem natural aplicados à identificação de discurso de ódio.

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., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Vargas, A. C. G., Paes, A., and Vasconcelos, C. N. (2016). Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In *Proceedings of the xxix conference on graphics, patterns and images*, volume 1. sn.