Identificação de Textos Ofensivos em Comentários do Instagram

Celso Luiz Silva Soares Filho[1]

[1]Programa de Pós-Graduação em Engenharia Elétrica – Universidade Federal do Maranhão (UFMA)
Av. dos Portugueses, 1966 - Vila Bacanga, São Luís - MA, 65080-805

celso.soares@discente.ufma.br

Abstract. *Offensive texts and hate speech are growing challenges on social media, especially in political discussions where divergent opinions often result in offenses or targeted attacks. This study proposes a method to identify offensive comments using the BERT model and the Hate Br Corpus, focusing on political posts on Instagram. By applying grid search for hyperparameter optimization, the model achieved remarkable results, with an F1-Score of 0.8649, Precision of 0.8657, and Recall of 0.8649. The analysis also demonstrated that preprocessing reduced the model's performance, highlighting the importance of preserving linguistic context when addressing this type of task. The study reinforces BERT's efficiency in detecting nuances in potentially offensive content, contributing to a more respectful digital environment.*

Resumo. *Textos ofensivos e discursos de ódio são desafios crescentes nas redes sociais, especialmente em discussões políticas, onde opiniões divergentes frequentemente resultam em ofensas ou ataques direcionados. Este trabalho propõe um método para identificar comentários ofensivos utilizando o modelo BERT e o Corpus Hate Br, focado em postagens políticas no Instagram. Com a aplicação de grid search para otimização dos hiperparâmetros, o modelo alcançou resultados expressivos, com F1-Score de 0.8649, Precisão de 0.8657 e Recall de 0.8649. A análise também demonstrou que o pré-processamento reduziu o desempenho do modelo, destacando a importância de preservar o contexto linguístico ao lidar com esse tipo de tarefa. O estudo reforça a eficiência do BERT na detecção de nuances em conteúdos potencialmente ofensivos, contribuindo para um ambiente digital mais respeitoso.*

1. Introdução

A ascensão da internet nos últimos anos tem provocado transformações significativas na vida das pessoas. Com ela, a comunicação tornou-se muito mais acessível e eficiente, permitindo que indivíduos ao redor do mundo troquem mensagens, fotos, vídeos e realizem chamadas de forma quase instantânea. Além disso, a internet possibilita o acesso a notícias de qualquer lugar e a qualquer momento, frequentemente por meio de redes sociais, que podem ser utilizadas em smartphones, tablets, computadores, entre outros dispositivos [OLIVEIRA et al. 2024].

Graças ao desenvolvimento de diversas redes sociais, como Instagram, X, Facebook, entre outras, as pessoas passaram a formar grupos e comunidades, compartilhando opiniões e sentimentos sobre interesses em comum [Arunachalam and Maheswari 2024].

A maneira mais comum de expressar essas opiniões é por meio de textos públicos, permitindo respostas e visualizações por um grande número de pessoas. Contudo, essa forma de exposição facilita a disseminação de conteúdos potencialmente ofensivos ou de ódio, que podem atingir grupos e causar ofensas ou constrangimentos.

Uma das áreas com maior volume de discussões e problemas relacionados a comentários ofensivos é a política. Muitas pessoas julgam e ofendem outras de forma desnecessária, apenas por discordarem de suas opiniões ou simplesmente por não apreciarem o conteúdo compartilhado [Grimminger and Klinger 2021]. Comentários ofensivos, criados apenas para atacar aqueles com opiniões divergentes, e discursos de ódio direcionados a grupos específicos — como manifestações racistas ou homofóbicas, por exemplo — são práticas que não apenas geram divisões, mas também prejudicam a convivência saudável nas plataformas digitais [Grimminger and Klinger 2021].

Nesse cenário, os algoritmos baseados em Large Language Models (LLM) tornam-se uma opção viável para a detecção de textos ofensivos e discursos de ódio nas redes sociais. Essas ferramentas visam promover um ambiente digital mais respeitoso, onde a liberdade de expressão possa ser exercida sem violar os direitos ou a dignidade de outras pessoas [Mozafari et al. 2022]. Entre os modelos mais utilizados para essa tarefa, destaca-se o BERT (Bidirectional Encoder Representations from Transformers), que, possui a capacidade de entender o contexto bidirecional das palavras em uma frase [Devlin 2018]. Isso permite identificar com precisão nuances linguísticas presentes em conteúdos ofensivos, tornando-o uma ferramenta eficiente para moderar interações online e auxiliar no combate a discursos prejudiciais.

Assim, esse trabalho possui o objetivo de propor um método de detecção de comentários ofensivos com BERT, avalaiando com a métrica f1-Score. O trabalho está organizado da seguinte forma: a Seção 2 fala sobre os trabalhos relacionados, que executam a classificação de textos ofensivo ou de discurso de ódio. A Seção 3 mostra os materiais e métodos utilizados para o desenvolvimento do trabalho. A Seção 4 Discute e expõe os resultados e experimentos realizados. Por fim, a Seção 5 mostra a conclusão e trabalhos futuros.

2. Trabalhos Relacionados

O trabalho desenvolvido por Vargas et al. (2021) montou um corpus com textos ofensivos e odiosos na língua portuguesa. O corpus foi criado utilizando comentários de postagens de diferentes políticos brasileiros, os quais foram anotados manualmente por especialistas. No final, obteve-se um total de 7.000 documentos anotados, sendo 3.500 comentários ofensivos e 3.500 não ofensivos. Foi realizada uma classificação do nível de ofensividade, podendo ser altamente, moderadamente ou levemente ofensivo. Além disso, foram identificados nove grupos de discurso de ódio: xenofobia, racismo, homofobia, sexismo, intolerância religiosa, partidarismo, apologia à ditadura, antissemitismo e gordofobia. Por fim, foram realizados alguns experimentos para detecção de ofensividade utilizando TF-IDF e SVM (Support Vector Machine), obtendo um f1-Score de 0,85. Para a detecção de discurso de ódio, alcançaram um f1-Score de 0,78, também utilizando TF-IDF, mas com Naive Bayes (NB) para a classificação.

No estudo de OLIVEIRA et al. (2024), foi apresentada uma abordagem para identificar discursos de ódio em textos extraídos de redes sociais. Além disso, analisou-se

como a distância léxica entre os idiomas dos corpora utilizados na pesquisa influencia os resultados. O trabalho também explorou o potencial do aprendizado interlinguístico (Cross-lingual Learning - CLL) como estratégia para aprimorar a identificação de discursos de ódio em diferentes línguas. Os experimentos realizados revelaram que a adoção do CLL elevou significativamente a eficácia dos modelos de classificação, alcançando um f1-score de 96,92%. Por fim, os autores concluíram que a diversidade linguística e a consideração da distância léxica em modelos baseados em Transformers desempenham um papel essencial na detecção de discursos de ódio.

No estudo conduzido por Assis et al. (2024), foi investigada a detecção de ódio em textos na língua portuguesa. Para isso, os autores analisaram modelos baseados em Transformers, avaliando diversas estratégias de treinamento e ativação. Foram examinados nove modelos distintos, variando em termos de arquitetura, tamanho e corpora utilizados no pré-treinamento. Os resultados indicaram que, embora grandes modelos generativos acessados por meio de prompts apresentem resultados promissores, modelos de linguagem de menor escala, devidamente ajustados, continuam a demonstrar um desempenho superior na execução dessa tarefa desafiadora.

O estudo de Grimminger and Klinger (2021) examinou como as campanhas eleitorais de 2020 nos Estados Unidos impactaram a comunicação online de apoiadores de Joe Biden e Donald Trump, investigando o uso de discursos odiosos e ofensivos. Para isso, foram anotados 3.000 tweets com base na detecção de discurso ofensivo e análise de postura, categorizando-os como favoráveis, contra, mistos, neutros ou sem expressão de opinião, além de avaliar o estilo ofensivo. Os resultados mostraram que identificar apoiadores de um candidato é altamente preciso (f1-Score de 0,89 para Trump e 0,91 para Biden), enquanto detectar oposição é mais difícil (f1-Score de 0,79 e 0,64, respectivamente). Já a identificação de discursos ofensivos apresentou maior desafio (f1-Score de 0,53).

Em Mozafari et al. (2022), foi identificado que a detecção de conteúdo abusivo online é desafiadora em línguas com poucos recursos devido à falta de dados rotulados e limitações dos modelos multilíngues. Para solucionar isso, foi proposta uma abordagem de meta-aprendizado, utilizando os modelos MAML e Proto-MAML. Experimentos realizados com 15 conjuntos de dados em 8 idiomas para discurso de ódio e 6 conjuntos de dados em 6 idiomas para linguagem ofensiva demonstraram que o meta-aprendizado supera o aprendizado por transferência na maioria dos casos. O Proto-MAML destacou-se, adaptando-se a novos idiomas com apenas 16 exemplos rotulados por classe.

3. Materiais e Método

Esta seção buscar explicar sobre o corpus utlizado, além de tecnicas e tecnologias aplicadas para a classficação, como o uso de pré-processamento, e a rede transformer, sendo a base para o BERT, o modelo utilizado.

3.1. Corpus

O corpus usado no trabalho foi o Hate Br, montado por Vargas et al. (2021), composto por 7.000 comentários de postagens no Instagram de contas relacionadas políticos. Os textos são rotulados em duas classes, entre textos não ofensivos e textos ofensivos anotadas por especialistas, divididos igualmente, ou seja 3.500 textos para cada classe. A Tabela 1 mostra exemplos dos textos do corpus.

Tabela 1. Exemplos de textos ofensivos e não ofensivos

Textos Ofensivos	Textos não ofensivos
Essa nao tem vergonha na cara!!	Vdd. Não ajuda em nada.
Essa mulher é doente.pilantra!	Comemorando a invasao da Polonia pelos nazistas do hitler???
Pena porque não vai mais roubar nosso Brasil, sua quenga velha	Já não bastava rebaixar os enfermeiros, agora isso!

3.2. Pré-Processamento

O pré-processamento de dados textuais é uma etapa essencial para garantir que os textos sejam adequadamente preparados antes de serem utilizados em modelos de Processamento de Linguagem Natural (NLP) [Navarro 2024]. Técnicas como remoção de emojis, números e pontuações, além de lematização, stemming e conversão para letras minúsculas, foram testadas neste trabalho com o objetivo de reduzir ruídos, padronizar os dados e melhorar o desempenho do modelo.

Essa etapa é crucial, pois textos brutos frequentemente contêm elementos que não contribuem diretamente para a tarefa de aprendizado, como emojis, números irrelevantes ou pontuações excessivas. A remoção de emojis, por exemplo, elimina símbolos que podem não agregar valor semântico em tarefas como a classificação de texto formal. Já a exclusão de números é especialmente relevante quando estes não possuem significado contextual, como em mensagens de redes sociais ou diálogos casuais.

Além disso, a padronização do texto, realizada por meio de técnicas como lematização, stemming e conversão para letras minúsculas, é fundamental para reduzir a complexidade do vocabulário. A lematização, ao reduzir palavras à sua forma base de maneira semântica, ajuda o modelo a generalizar melhor o significado das palavras. O stemming, por sua vez, simplifica ainda mais os tokens ao remover sufixos. Por fim, a conversão para letras minúsculas garante uniformidade, eliminando redundâncias causadas por diferenças de capitalização.

3.3. Transformer

Os Transformers foram introduzidos no artigo “Attention is All You Need”, por Vaswani et al. (2017). Diferentemente de arquiteturas anteriores, como as Redes Recorrentes (RNNs), os Transformers utilizam o mecanismo de atenção como peça central, permitindo que o modelo processe e relacione informações textuais sem depender de uma sequência fixa. Essa abordagem tornou os Transformers essenciais para tarefas como tradução automática, geração de texto e classificação.

Na Figura 1, é apresentada a arquitetura Transformer, que consiste em dois componentes principais: o codificador e o decodificador. O codificador processa a entrada e gera representações intermediárias, enquanto o decodificador utiliza essas representações para gerar a saída, como uma tradução ou resposta a uma pergunta. Cada bloco desses componentes é composto por camadas de atenção e subcamadas totalmente conectadas, que operam em paralelo, proporcionando maior eficiência e escalabilidade [Vaswani 2017].

O mecanismo de atenção é uma parte muito importante da arquitetura, projetado para superar as limitações das arquiteturas sequenciais. Ele permite que o modelo fo-

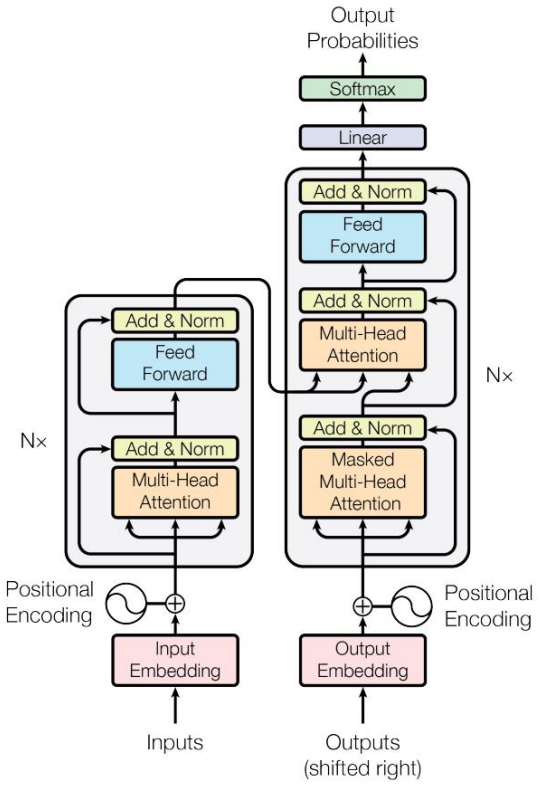

Figura 1. Representação de entrada do BERT. Fonte: [Vaswani 2017]

que em partes específicas da entrada ao processar cada palavra, aprendendo as relações contextuais entre os elementos do texto. Isso é alcançado analisando toda a sequência simultaneamente, atribuindo pesos diferentes às palavras com base em sua relevância [Vaswani 2017].

Além disso, o Multi-Head Attention é uma extensão do mecanismo de atenção que aumenta a capacidade do modelo de capturar múltiplas relações contextuais ao mesmo tempo [Vaswani 2017]. Em vez de calcular a atenção uma única vez, o Multi-Head Attention concatena as saídas de várias cabeças de atenção, permitindo que o modelo capture diferentes aspectos das relações contextuais, como informações gramaticais e semânticas [Vaswani 2017].

3.4. BERT

Para realizar a tokenização dos textos e a classificação, foi utilizado o modelo BERT (Bidirectional Encoder Representations from Transformers), proposto por Devlin et al. (2018). Esse modelo permite trabalhar com uma abordagem bidirecional, possibilitando a compreensão do contexto das palavras considerando simultaneamente os textos que as antecedem e sucedem [Devlin 2018]. O BERT foi escolhido devido à sua capacidade de aprendizado contextual profundo, alcançada por meio de uma combinação de técnicas avançadas de tokenização, embeddings enriquecidos e uma arquitetura robusta.

Na arquitetura Transformer, é necessário transformar os textos em tokens para que possam ser processados [Vaswani 2017]. No BERT, o processo de tokenização fragmenta palavras em subpalavras menores. Essa técnica lida eficientemente com palavras raras,

já que qualquer palavra pode ser representada por uma combinação de tokens existentes no vocabulário [Devlin 2018]. Após a tokenização, cada token recebe uma representação vetorial composta por três embeddings: de palavra, que representa o significado do token; posicional, que indica a posição do token na sequência; e de segmento, que distingue diferentes partes do texto (como em tarefas de pares de perguntas e respostas), conforme ilustrado na Figura 2.

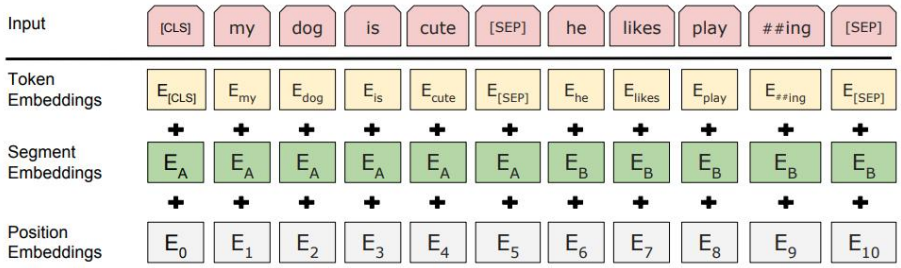

Figura 2. Representação de entrada do BERT. Fonte: [Devlin 2018]

O maior diferencial do BERT está na sua bidirecionalidade, alcançada por meio das camadas de atenção da arquitetura Transformer [Devlin 2018]. Além disso, é possível realizar o Fine-Tuning, um processo em que uma camada de classificação linear é adicionada ao topo do modelo BERT pré-treinado. O token especial [CLS], que representa toda a sequência de entrada, passa por essa camada para prever a classe (ofensivo ou não ofensivo) [Devlin 2018]. Durante o treinamento, os pesos do modelo são ajustados com base em um conjunto de dados rotulado, permitindo que ele seja refinado para a tarefa específica, sem perder os conhecimentos adquiridos no pré-treinamento.

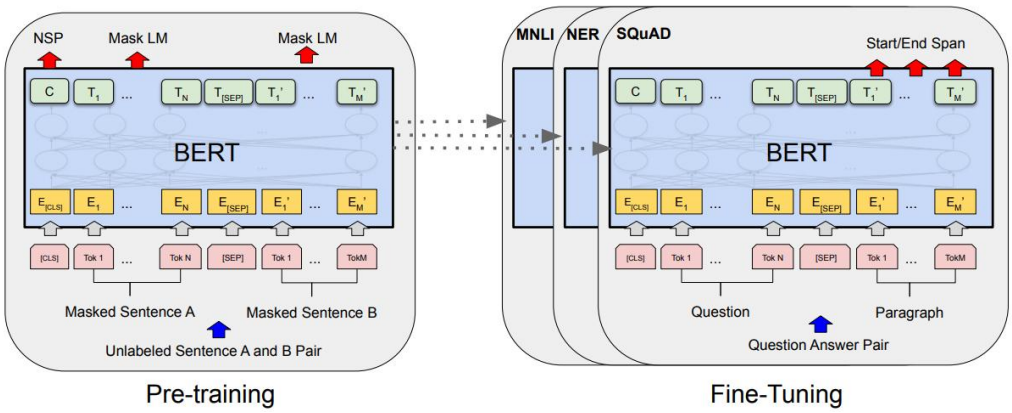

Figura 3. Representação de entrada do BERT. Fonte: [Devlin 2018]

Na Figura 3, é possível observar que o BERT é pré-treinado com grandes conjuntos de dados, utilizando a técnica de mascarar certas palavras na entrada e analisar o contexto das palavras anteriores e seguintes. Isso permite ao modelo capturar um entendimento contextual abrangente da sentença. Esse modelo pré-treinado pode ser reutilizado e ajustado posteriormente por meio de Fine-Tuning, o que possibilita ao modelo alcançar melhores resultados em conjuntos de dados e tarefas específicas.

4. Expexrimentos e Resultados

nicialmente, o corpus foi dividido entre treino, validação e teste. Como resultado, obteve-se 5.040 textos para treino (80%), 560 para validação (10% do treino) e 1.400 amostras para teste (20%), com distribuição igual entre as classes em cada conjunto.

O primeiro experimento foi realizado apenas com o fine-tuning do modelo BERT. Posteriormente, com o objetivo de melhorar os resultados das métricas de avaliação, foi aplicado um pré-processamento nos textos, incluindo remoção de emojis, números, pontuações, lematização, stemming e a conversão das letras para minúsculas. Essas etapas resultaram em uma melhora nas métricas. A Tabela 2 apresenta os resultados antes e depois do pré-processamento nos textos, antes do fine-tuning.

Ambos os experimentos foram realizados utilizando os seguintes hiperparâmetros: 3 épocas, batch size de 16, taxa de aprendizado de 1e-4, otimizador AdamW e a função de perda integrada do modelo BERT.

Tabela 2. Resultados com e sem pré-processamento nos textos

Experimentos	F1-Score	Precisão	Recall
Sem Pré-processamento	0.42	0.74	0.54
Com Pré-processamento	0.35	0.68	0.51

Como os testes foram realizados sob as mesmas condições, constatou-se que a aplicação de pré-processamentos nos textos prejudicava os resultados, uma vez que poderia modificar informações críticas que o modelo foi projetado para interpretar diretamente [Kancharapu and Ayyagari 2024].

Uma alternativa adotada para melhorar o desempenho do modelo de classificação foi a utilização do Grid Search, com o objetivo de testar diversos hiperparâmetros e identificar a melhor combinação que proporcionasse o maior valor de F1-Score [Bergstra and Bengio 2012].

A Tabela 3 apresenta os hiperparâmetros testados no Grid Search, destacando em negrito aqueles que obtiveram os melhores resultados.

Tabela 3. Opções de hiperparâmetros para o Grid Search

Hiperparâmetros	Opção 1	Opção 2	Opção 3
Taxa de aprendizado	1e-5	**3e-5**	5e-5
Otimizadores	Adam	SGD	**AdamW**
Função de Perda	**Cross Entropy**	MSE Loss	-
Batch size	**8**	16	32
Épocas	3	4	**5**

Ao realizar um novo experimento com os resultados do Grid Search, foi possível melhorar significativamente os resultados, como pode ser observado na Tabela 4.

Em comparação com os trabalhos encontrados na literatura que realizam o mesmo tipo de tarefa, a classificação feita no presente estudo, por se tratar de um método simples e aplicável apenas a um corpus em língua portuguesa, apresentou desempenhos que não

Tabela 4. Resultados dos experimentos após o Grid Search

F1-Score	Precisão	Recall
0.8714	0.8717	0.8714

foram tão satisfatórios. Contudo, o método superou o trabalho de Vargas et al. (2021), que propôs o corpus e realizou a classificação utilizando TF-IDF e SVM (Support Vector Machine). A Tabela 5 apresenta a comparação dos resultados obtidos no presente trabalho com outros da literatura.

Tabela 5. Comparação com Resultados encontrados na Literatura

Método	F1-Score	Precisão	Recall
BERT [Wadud et al. 2023]	-	93%	-
TF-IDF / SVM [Garcıa and Bedmar 2021]	-	89%	-
LSTM-BOOST [Wadud et al. 2022]	89%	92%	-
TF-IDF / SVM [Vargas et al. 2021]	85%	-	-
Método proposto	0.8714	0.8717	0.8714

5. Conclusão

O presente trabalho buscou realizar uma classificação binária em comentários retirados de postagens de políticos na rede social Instagram. Para isso, foram utilizados o Corpus Hate Br e o modelo BERT. No final, foi possível alcançar um F1-Score de 0,8649, uma Precisão de 0,8657 e um Recall de 0,8649.

As métricas também foram analisadas com o uso de pré-processamento, que, no entanto, acabou prejudicando o desempenho. Com o objetivo de melhorar os resultados, foi utilizado o Grid Search, permitindo encontrar a melhor combinação entre os hiperparâmetros definidos, o que resultou nos melhores resultados da pesquisa.

Por fim, os resultados foram comparados com outros estudos que também buscaram realizar a classificação de textos ofensivos. Apenas o trabalho de Vargas et al. (2021) utilizou o mesmo Corpus para classificação. Para trabalhos futuros, espera-se empregar o BERT como tokenizador, mas realizar a classificação com outros algoritmos, como a máquina de vetores de suporte (SVM), que é comumente utilizada.

Referências

Arunachalam, V. and Maheswari, N. (2024). Enhanced detection of hate speech in dravidian languages in social media using ensemble transformers. *Interdisciplinary Journal of Information, Knowledge, and Management*, 19:036.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).

Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Garcıa, M. N. and Bedmar, I. S. (2021). Detecting offensiveness in social network comments. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021), CEUR Workshop Proceedings. CEURWS. org*.

Grimminger, L. and Klinger, R. (2021). Hate towards the political opponent: A twitter corpus study of the 2020 us elections on the basis of offensive speech and stance detection. *arXiv preprint arXiv:2103.01664*.

Kancharapu, R. and Ayyagari, S. N. (2024). Depression detection: Unveiling mental health insights with twitter data and bert models.

Mozafari, M., Farahbakhsh, R., and Crespi, N. (2022). Cross-lingual few-shot hate speech and offensive language detection using meta learning. *IEEE Access*, 10:14880–14896.

Navarro, G. (2024). Fair and ethical resume screening: Enhancing ats with justscreen the resumescreeningapp. *Journal of Information Technology, Cybersecurity, and Artificial Intelligence*, 2(1):1–7.

OLIVEIRA, A. B. d. et al. (2024). Detecção de discurso de ódio em comentários relacionados à política.

Vargas, F. A., Carvalho, I., de Góes, F. R., Benevenuto, F., and Pardo, T. A. S. (2021). Hatebr: A large expert annotated corpus of brazilian instagram comments for offensive language and hate speech detection. *arXiv preprint arXiv:2103.14972*.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Wadud, M. A. H., Kabir, M. M., Mridha, M. F., Ali, M. A., Hamid, M. A., and Monowar, M. M. (2022). How can we manage offensive text in social media-a text classification approach using lstm-boost. *International Journal of Information Management Data Insights*, 2(2):100095.

Wadud, M. A. H., Mridha, M. F., Shin, J., Nur, K., and Saha, A. K. (2023). Deepbert: Transfer learning for classifying multilingual offensive texts on social media. *Computer Systems Science & Engineering*, 44(2).