Detecção de Cyberbullying com Abordagens de Deep Learning em uma Base Multiclasse

Detection of Cyberbullying with Deep Learning Approaches in a Multiclass DataBase

José Carvalho[1]
orcid.org/0009-0005-0535-3301

Roberta Andrade de Araújo Fagundes[2]
orcid.org/0000-0002-7172-4183

[1]Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil.
E-mail:jaec@ecomp.poli.br

[2]Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil.
E-mail: roberta.fagundes@upe.br

DOI: 10.25286/repa.v11i4.3973

RESUMO

O *cyberbullying* caracteriza-se por comportamentos agressivos e repetitivos em ambientes digitais, com o objetivo de infligir danos psicológicos e emocionais através de humilhações, ameaças e difamações. Dado o volume massivo de dados gerados *online*, a moderação humana torna-se inviável devido a limitações de escalabilidade, custos e sobrecarga cognitiva, tornando indispensável o desenvolvimento de ferramentas de detecção automatizada. Embora modelos de Aprendizado de Máquina (AM) sejam eficazes na identificação de padrões textuais, a variabilidade da linguagem natural (e.g., gírias, abreviações) impõe desafios persistentes. Esta pesquisa investiga abordagens para mitigar tais desafios em uma base de dados multiclasse, avaliando o desempenho de métodos de *word embedding* (Word2Vec e FastText) integrados a algoritmos de aprendizado profundo (LSTM e GRU), com a incorporação de mecanismos de atenção e camadas bidirecionais. A avaliação utilizou métricas de acurácia, precisão, *recall*, F1-Score e análise de matriz de confusão. Os resultados indicam que, para vetores Word2Vec, o modelo BiGRU com atenção obteve o melhor desempenho (Precisão: 82%; *Recall*: 80,5%; F1-Score: 80,3%; Acurácia: 81%). Em contrapartida, com o uso de FastText, o modelo BiLSTM apresentou os melhores resultados (Precisão: 81,3%; *Recall*: 79,2%; F1-Score: 79%; Acurácia: 79%).

PALAVRAS-CHAVE: *Cyberbullying*, Aprendizado de Máquina, multiclasse

ABSTRACT

Cyberbullying is characterized by repetitive aggressive behaviors in digital environments, aimed at inflicting psychological and emotional harm through humiliation, threats, and defamation. Given the massive volume of user-generated content, human moderation is insufficient due to scalability issues and high operational costs, necessitating the development of automated detection tools. While Machine Learning (ML) models are powerful for detecting intrinsic textual patterns, they face significant challenges regarding linguistic variability (e.g., slang, abbreviations). This study explores approaches to mitigate these challenges within a multi-class cyberbullying dataset, evaluating word embedding methods (Word2Vec and FastText) integrated with Deep Learning algorithms (LSTM and GRU), utilizing architectural modifications such as attention mechanisms and bidirectional layers. Models were assessed using accuracy, precision, recall, F1-Score, and confusion matrix analysis. Results indicate that for Word2Vec embeddings, the BiGRU model with attention achieved the best performance (Precision: 82%; Recall: 80.5%; F1-Score: 80.3%; Accuracy: 81%). Conversely, for FastText, the BiLSTM model yielded the highest results (Precision: 81.3%; Recall: 79.2%; F1-Score: 79%; Accuracy: 79%).

KEYWORDS: Cyberbullying, Machine Learning, multiclass

1 INTRODUÇÃO

O *bullying*, historicamente confinado a escolas e outros ambientes sociais, assumiu uma dimensão nova e alarmante com ampla adoção das redes sociais e o acesso ubíquo à internet. Essa evolução deu origem ao *cyberbullying*, definido como comportamento deliberado e agressivo disseminado por plataformas digitais, como aplicativos de mensagens, redes sociais e fóruns online, com a intenção de causar dano psicológico, emocional ou até físico. Ao contrário do *bullying* tradicional, o *cyberbullying* pode ocorrer anonimamente e escalar rapidamente, intensificando seu alcance e consequências. Vítimas, especialmente adolescentes e jovens adultos, frequentemente sofrem de humilhação, ameaças e difamação, o que pode levar a trauma de longo prazo, ansiedade, depressão e até suicídio **[1]**.

A gravidade do *cyberbullying*, aliada à dificuldade de monitoramento manual, impõe a necessidade urgente de mecanismos de detecção automatizada que operem em tempo real, viabilizando a intervenção imediata. A moderação humana tradicional tornou-se inviável, uma vez que o volume e a velocidade de produção de conteúdo digital excedem a capacidade de processamento manual. Além da impossibilidade de escala, a moderação humana apresenta desvantagens críticas, como alto custo operacional, cansaço e suscetibilidade a vieses. Consequentemente, o desenvolvimento de abordagens computacionais robustas consolidou-se como um requisito indispensável para a segurança em ambientes virtuais.

Avanços em inteligência artificial, especialmente *Machine Learning* (ML) e Processamento de Linguagem Natural (PLN), oferecem mecanismos escaláveis e adaptáveis para identificar automaticamente padrões característicos dos textos que possuem *cyberbullying*, possibilitando o bloqueio de conteúdo ofensivo e o alerta de moderados para a proteção das vítimas **[2]**.

Apesar de seu potencial, a detecção de *cyberbullying* permanece um desafio significativo, dada a natureza informal, idiomática e altamente contextual da linguagem *online*. Motivado por esses problemas encontrados, este estudo focou em analisar abordagens de *Deep Learning* (DL) para analisar e detectar *cyberbullying* em uma base textual multiclasse; modificar as arquiteturas dos modelos *Long-Short Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU) com à adição de camadas bidirecionais e de atenção, e diferentes tipos de modelos de *word embedding* (word2vec e fasttext), com o objetivo de criação de vetores mais robustos de entrada para os modelos de DL.

2 OBJETIVO

O presente trabalho se objetiva no contexto da detecção de *cyberbullying* em ambientes virtuais. A exploração de diferentes abordagens e ambientes metodológicos possibilitará *insights* valiosos sobre a robustez e a generalização dos modelos, contribuindo significativamente para estudos futuros. Além disso, esta pesquisa busca alternativas aos modelos classificados como Estado da Arte, especialmente aqueles baseados na arquitetura de *Transformers* (e.g., BERT, RoBERTa). Embora estes modelos atinjam o melhor desempenho, o seu alto custo computacional justifica a busca por soluções mais eficientes e com menor demanda de recursos que as arquiteturas de maior complexidade paramétrica.

3 REVISÃO BIBLIOGRÁFICA

Esta seção é dedicada à apresentação do arcabouço teórico que fundamenta esta pesquisa.

3.1 Cyberbullying

O cyberbullying constitui um problema de crescente relevância, impactando a vida de milhares de indivíduos, com especial incidência entre jovens e adolescentes. Este conceito representa uma evolução do bullying tradicional, que historicamente se manifesta por ataques verbais e físicos para o ambiente digital. Essa transição para os meios eletrônicos tem contribuído para um aumento no número de vítimas que desenvolvem problemas psicológicos, físicos e comportamentais **[3]**.

3.2 Redes Neurais Recorrentes

As Recurrent Neural Networks (RNN) constituem uma classe de algoritmos de DL projetadas, em sua essência, para lidar com dados sequenciais. A principal característica das RNNs reside em sua capacidade de manter uma memória interna de entradas passadas, o que lhes permite processar sequências de dados de forma eficaz. Essa propriedade as torna ideais para aplicações em PLN, reconhecimento de fala e predição de séries temporais, onde o contexto e a ordem dos dados são cruciais **[4]**. Dentre as RNNs mais proeminentes, destacam-se a Long Short-Term Memory (LSTM) e sua versão simplificada, a Gated

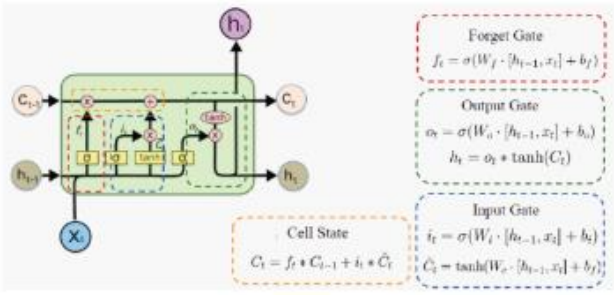
Recurrent Unit (GRU).

3.2.1 LSTM

As LSTMs, uma arquitetura mais avançada das RNNs, superam essa limitação ao aumentar sua capacidade de armazenamento de informações de longo prazo, o que as torna amplamente aplicáveis em tarefas como análise textual e séries temporais. As células LSTM são capazes de extrair informações, lembrar de informações ou deletar informações armazenadas que não são mais necessárias **[5]**.

A estrutura padrão do LSTM pode ser observada na Figura 1, no qual é constituída por: porta de entrada (*input gate*), responsável pela leitura dos vetores de característica, a célula de estado (*cell state*), que conserva os valores de entrada, a porta de saída (*output gate*), responsável pela escrita do valor final, e porta de esquecimento (*forget gate*), responsável por deletar informações que não são mais necessárias **[7]**.

Figura 1 - Arquitetura padrão do LSTM



Fonte: [6].

3.2.2 GRU

A GRU é outro tipo de RNN, criada como uma alternativa mais simplificada e dinâmica à arquitetura LSTM. Diferentemente da LSTM, que emprega três portas (input, output e forget), o modelo GRU opera com apenas duas portas: a porta de atualização (update gate) e a porta de reinicialização (reset gate).

3.3 Vetorização dos Dados

A etapa de vetorização dos dados, a informação em linguagem natural é transformada em vetores numéricos, formato compreensível para análise computacional. Embora exista uma vasta gama de técnicas de vetorização, esta pesquisa concentrou-se naquelas que oferecem representações mais detalhadas dos dados e que são amplamente empregadas na literatura científica.

3.3.1 Word2Vec

Word2Vec é um modelo que emprega o método de embedding para gerar vetores de palavras. Este método é capaz de criar representações vetoriais densas e detalhadas que capturam as relações semânticas e contextuais entre as palavras. Consequentemente, os dados de entrada fornecidos aos modelos de *Machine Learning* (ML) tornam-se mais informativos **[8]**.

O treinamento de um modelo Word2Vec envolve dois passos principais: A utilização de um modelo *Continuous Bag-of-Words* (CBOW) para aprender vetores de palavras contínuas (*continuous word vectors*). Essas são representações numéricas onde palavras com significados ou contextos similares possuem representações numéricas também similares; O treinamento de um modelo de linguagem baseado em redes neurais **[9]**.

3.3.2 FastText

FastText é um modelo de embedding de palavras que se distingue por sua capacidade de incorporar informações de subpalavras através do uso de recursos de n-gramas. Ele integra técnicas de redução de dimensionalidade e emprega uma aproximação eficiente da função softmax, o que resulta em um treinamento e classificação mais rápidos. O modelo utiliza a função softmax para calcular a distribuição discreta de probabilidade sobre as classes predefinidas. Para um conjunto de N documentos, este processo otimiza a minimização do log-likelihood negativo sobre as classes **[10, 11]**.

4 METODOLOGIA

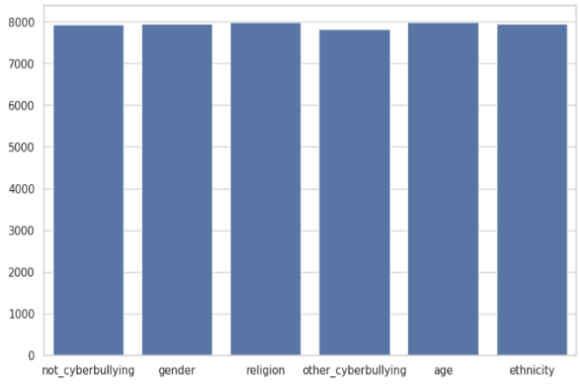
Esta seção tem como objetivo descrever a metodologia empregada pela pesquisa. Ela foi baseada pelas fases da framework da *CRoss-Industry Standard Process for Data Mining* (CRISP-DM) **[12]** e como o objetivo da pesquisa não envolvia a criação de um produto, as etapas adotadas foram: entendimento dos dados, pré-processamento, modelagem e avaliação.

4.1 Entendimento dos Dados

DOI: 10.25286/repa.v11i4.3973

Esta seção aborda a fase de entendimento dos dados da pesquisa, que consiste em uma descrição aprofundada da base de dados central para o estudo. Considerando a vasta quantidade de dados acessíveis em inglês, uma base de dados específica foi obtida na plataforma Kaggle. Esta base consiste em um total de 47.656 entradas não duplicadas (as linhas duplicadas foram removidas utilizando a biblioteca Pandas) e duas colunas: uma contendo o texto dos tweets e a outra a classificação correspondente ao tipo de *cyberbullying*. Os tipos de *cyberbullying* são: gênero (*gender*), etnia (*ethnicity*), idade (*age*), religião (*religion*), outros *cyberbullying* (other *cyberbullying*) e não cyberbullying (*not cyberbullying*). A Figura 2 ilustra a quantidade de instância em cada uma das classes.

Figura 2 - Quantidade de instâncias por classe



Fonte: Autor (2025)

4.1 Pré-processamento

Esta etapa é crucial para preparar os dados a serem analisados pelos modelos. Frequentemente, os conjuntos de dados brutos contêm informações irrelevantes que, além de serem desnecessárias, aumentam significativamente o custo computacional. Além disso, existe também a necessidade de transformar a informação em linguagem natural em vetores de números, formato que é compreensível para os algoritmos. As etapas realizadas foram: a limpeza da informação, a utilização de técnicas de PLN e a vetorização da informação (explicada na seção 3.3).

4.1.1 Limpeza dos Dados

A limpeza da base de dados é uma etapa crucial para otimizar o processamento e reduzir o custo computacional. As principais operações de limpeza realizadas incluíram: a remoção de comentários (texto após o símbolo '#'), emojis, citações de usuários (texto após o símbolo '@'), pontuações, símbolos e números. Somado ao que já foi descrito, todas as letras maiúsculas foram convertidas para minúsculas.

4.2.2 PLN

As técnicas de PLN, ou Natural Language Processing (NLP), são cruciais para a preparação dos dados antes da análise pelos modelos de ML. As técnicas aplicadas neste estudo, utilizando a biblioteca Natural Language Toolkit (NLTK), incluem: A tokenização que é o processo de segmentação de frases em unidades menores, como palavras ou termos (tokens). A remoção de stopwords, processo de exclusão de palavras gramaticais comuns (e.g., artigos, preposições, advérbios) que, isoladamente, não contribuem significativamente para a semântica da análise. E a lematização que reduz as flexões verbais e nominais das palavras à sua forma base ou lema **[13]**.

4.2.3 Vetorização dos Dados

Nesta etapa, a informação em linguagem natural é transformada em vetores numéricos, formato compreensível para análise computacional. Como descrito, serão aplicados os modelos de *word embedding* (Word2Vec e FastText) capazes de criar representações vetoriais densas e detalhadas que capturam as relações semânticas e contextuais entre as palavras **[14]**.

4.3 Modelagem

Esta etapa é responsável por descrever os modelos que foram construídos pela pesquisa. Foram escolhidos dois dos principais modelos de RNNs encontrados na literatura que foram o LSTM e GRU. Além das arquiteturas de modelo base, foram incorporadas camadas bidirecionais e de atenção aos modelos LSTM e GRU.

A camada de atenção, em particular, é responsável por ponderar a importância de diferentes partes dos dados de entrada, destacando os elementos mais relevantes para a análise **[15]**.

A ideia sobre o uso da camada bidirecional é a construção do modelo com duas redes recorrentes (LSTM ou GRU), ambas conectadas à mesma camada

de saída, que irão receber a sequência de treinamento, lendo-a da esquerda para a direita e da direita para a esquerda. Isso significa que para cada ponto na sequência, a RNN tem informações completas anteriores e posteriores ao ponto de interesse **[16]**.

4.4 Avaliação

Esta etapa é responsável por avaliar as performances dos modelos a partir de métricas; as métricas utilizadas foram a matriz de confusão, acurácia, precisão, recall e F1-Score. De acordo com Geron **[17]**:

- A matriz de confusão é utilizada para acessar o número de resultados preditos pelo modelo de ML, comparando os resultados com os valores atuais;
- Acurácia representa a proporção de predições globalmente corretas (tanto Positivas quanto Negativas) em relação ao conjunto total de instâncias. Embora seja uma medida geral de desempenho, pode ser enganosa em conjuntos de dados desbalanceados;
- Precisão mede a proporção de predições Positivas que estavam de fato corretas. É uma métrica de quão confiável é o modelo ao predizer a classe Positiva;
- Recall mede a proporção de todas as instâncias que eram realmente Positivas que o modelo conseguiu identificar corretamente. É uma métrica da capacidade do modelo de predizer todos os casos Positivos;
- F1-Score é a média harmônica entre a Precisão e o Recall, fornecendo um único valor que equilibra ambas as métricas. É particularmente útil em cenários com classes desbalanceadas, onde tanto a minimização de Falsos Positivos (visada pela Precisão) quanto a de Falsos Negativos (visada pelo Recall) são importantes.

5. RESULTADOS

Esta seção é responsável por descrever os resultados obtidos pela pesquisa. Os subtópicos serão separados com relação ao word2vec e fasttext. As saídas geradas pelos métodos de *word embedding* servirão de entrada aos modelos (LSTM e GRU) com suas variações (com camada de atenção e bidirecional).

Os modelos foram estruturados com a seguinte arquitetura: uma camada de entrada, uma camada *embedding*, uma camada de dropout, uma camada de atenção, porém, também foi testada a sua ausência, duas camadas empilhadas de LSTM ou GRU, sendo também avaliadas suas versões bidirecionais, uma camada totalmente conectada (*Dense Layer*) e, por fim, uma camada de saída.

Os hiperparâmetros utilizados para os modelos LSTM e GRU foram definidos da seguinte forma: a dimensão dos vetores de entrada foi de 300, o comprimento máximo das sequências foi estabelecido em 50 tokens e a taxa de *dropout* foi fixada em 0,5. O número de unidades ocultas (output space) nas camadas LSTM e GRU foi de 256, enquanto a camada totalmente conectada (Dense) apresentou 128 unidades com a função de ativação ReLU. A camada de saída teve seu número de unidades correspondente à quantidade de classes do conjunto de dados e utilizou a função de ativação *softmax* para a geração das probabilidades preditivas.

Os modelos LSTM e GRU foram configurados para serem executados em 25 épocas, com a implementação de um mecanismo parada antecipada, e com *batch size* de 128. A base de dados foi separada na proporção treino, validação e teste igual a 7:1:1.

5.1 Word2Vec

Esta seção será responsável por descrever uma análise quali-quantitativa dos resultados obtidos pela matriz de confusão e as métricas de avaliação (acurácia, precisão, *recall* e *F1-Score*).

A matriz de confusão (Figura 3) apresenta o desempenho de classificação de um dos modelos empregando o método *word2vec*, com um padrão de comportamento semelhante sendo observado nas demais arquiteturas. A eficácia geral é alta, conforme indicado pela diagonal principal bem definida, que representa as predições corretas.

Contudo, observa-se uma maior ambiguidade de classificação (mistura de cores) nas linhas (valores verdadeiros) correspondentes às classes *gender*, *other cyberbullying* e *not cyberbullying*. Analisando a distribuição dos erros de classificação (valores fora da diagonal), nota-se que as classes other *cyberbullying* e *not cyberbullying* são as principais fontes de confusão, impactando diretamente a eficiência dos modelos. Em termos numéricos, a confusão se distribui da seguinte forma:

- Classe *gender* (Verdadeira)**:** O modelo previu 160 instâncias como *not cyberbullying* e 300 como *other cyberbullying*.

DOI: 10.25286/repa.v11i4.3973

- Classe *not cyberbullying* (Verdadeira): O modelo previu 620 instâncias incorretamente como *other cyberbullying*.
- Classe *other cyberbullying* (Verdadeira): O modelo previu 560 instâncias incorretamente como *not cyberbullying*.

Essa distribuição de erros demonstra a dificuldade dos modelos em distinguir entre instâncias de *not cyberbullying* e as classificadas como *other cyberbullying*, o que sugere uma alta ambiguidade semântica entre esses dois rótulos no *corpus*.

O resultado obtido, que aponta para uma ambiguidade na classificação entre instâncias específicas, sublinha a dificuldade inerente à distinção desses rótulos. Dada a inviabilidade de escalabilidade e o alto custo operacional associados à anotação manual de *datasets* extensos (≃50 mil linhas), que também está sujeita a erros humanos e vieses, torna-se imperativo que trabalhos futuros explorem abordagens de aprendizado semi-supervisionado ou não supervisionado.

Tais técnicas podem oferecer uma solução eficiente para a melhor separação e distinção das instâncias, mitigando o esforço de rotulagem e a dependência de grandes volumes de dados anotados manualmente para o treinamento.

A Tabela 1 detalha a média das métricas de desempenho (acurácia, precisão, *recall* e *F1-Score*) para os modelos LSTM e GRU, incluindo suas variações bidirecionais (Bi) e com mecanismo de atenção (A).

A adição da camada bidirecional (BiLSTM) proporcionou melhorias notáveis nas métricas de precisão e *recall* em relação ao modelo LSTM unidirecional.

A inclusão do mecanismo de atenção (BiLSTM-A) resultou em um *F1-Score* superior, indicando um melhor equilíbrio entre precisão e *recall* para o modelo. No entanto, o BiLSTM-A apresentou uma precisão ligeiramente inferior se comparado à arquitetura BiLSTM sem atenção.

No modelo GRU, a implementação da bidirecionalidade (BiGRU) proporcionou melhores resultados nas métricas de acurácia e *F1-Score*, superando os modelos unidirecionais.

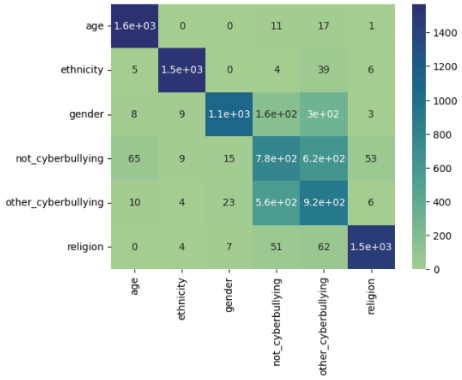
A camada de atenção (BiGRU-A) demonstrou um impacto significativo no *recall*, elevando a capacidade do modelo de identificar corretamente as instâncias de cyberbullying (positivos verdadeiros).

Tabela 1 - Resultado Médio dos Modelos com Word2Vec

Precisão	Recall	F1-Score	Acurácia	Modelos
82	80,3	80,8	80	BiLSTM (A)
81,3	79,3	79,8	80	LSTM (A)
82,3	80,3	80,2	81	BiLSTM
80,5	79,7	80	80	LSTM
82	80,5	80,3	81	BiGRU (A)
82	79,8	80,2	80	GRU (A)
82	80,3	80,7	81	BiGRU
82,5	80,3	80	80	GRU

Fonte: Autor (2025)

Figura 3 - Matriz de Confusão com Word2Vec



Fonte: Autor (2025)

5.2 FastText

Esta seção apresenta a avaliação de desempenho dos modelos utilizando o FastText como método de *word embedding*. Seguindo a metodologia aplicada ao *Word2Vec*, realiza-se uma análise quali-quantitativa. No entanto, visando a otimização computacional, a etapa quantitativa restringiu-se aos modelos equipados com mecanismo de atenção, comparando especificamente o impacto da arquitetura bidirecional *versus* unidirecional.

A Figura 4 ilustra a matriz de confusão representativa para os modelos baseados em FastText, cujo padrão de comportamento assemelha-se ao observado nas demais arquiteturas. Semelhante aos resultados com *Word2Vec*, nota-se uma diagonal principal bem definida, evidenciando a eficácia geral da classificação. Entretanto, persiste uma confusão significativa (alta correlação) entre as classes other

cyberbullying, not cyberbullying e gender.

A recorrência deste comportamento corrobora a hipótese de que o desafio de classificação reside na complexidade intrínseca da base de dados, sublinhando a necessidade de investigar modelos com maior capacidade de discriminação de características entre estas classes específicas.

A Tabela 2 apresenta os valores médios das métricas de desempenho (Acurácia, Precisão, *Recall* e F1-Score) para os modelos LSTM e GRU. A incorporação da camada bidirecional promoveu incrementos na precisão e no *recall*, aprimorando a capacidade do modelo em identificar verdadeiros positivos. Contudo, observou-se uma ligeira redução no F1-Score, enquanto a acurácia permaneceu inalterada em comparação à versão unidirecional.

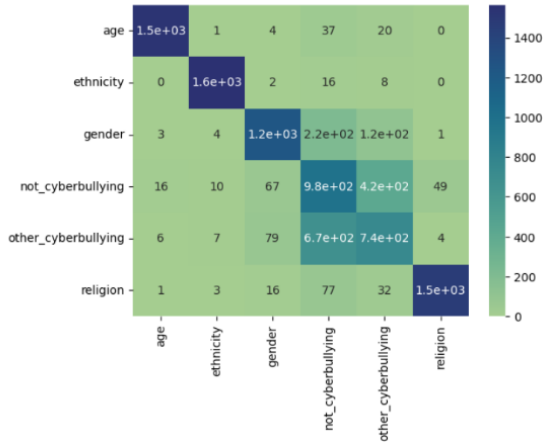
No modelo GRU, a bidirecionalidade demonstrou-se mais benéfica, impulsionando o desempenho em três das quatro métricas avaliadas (Recall, F1-Score e Acurácia). A precisão foi a única métrica que não registrou ganho com esta alteração arquitetural.

Tabela 2 - Resultado Médio dos Modelos com FastText

Precisão	Recall	F1-Score	Acurácia	Modelos
81,3	79,2	79	79	BiLSTM
79,8	78,7	79,2	79	LSTM
79,7	78,3	78,5	79	BiGRU
79,7	77,8	77,7	78	GRU

Fonte: Autor (2025)

Figura 4 - Matriz de Confusão com FastText



Fonte: Autor (2025)

6 CONCLUSÕES

Nesta seção serão discutidas as conclusões alcançadas a partir dos resultados da pesquisa, divididas nos seguintes subtópicos: Eficiência e Desempenho; Análise de Erros e Limitações; Relevância da Pesquisa; e Trabalhos Futuros.

A aplicação de modelos de detecção automática de cyberbullying em ambientes virtuais representa um avanço essencial para a promoção de interações mais seguras e saudáveis nas plataformas digitais. Ao permitir a identificação rápida e precisa de comportamentos ofensivos. Nesse sentido, o presente trabalho contribui para ampliar o acesso a tecnologias de monitoramento inteligente, reforçando a importância de modelos que conciliam desempenho, robustez e viabilidade operacional.

Os experimentos demonstraram que o método Word2Vec conferiu maior robustez aos modelos em comparação ao FastText. Além da qualidade das predições, o Word2Vec mostrou-se computacionalmente mais eficiente, reduzindo o tempo de treinamento pela metade (260s por época) em comparação aos 520s exigidos pelo FastText.

A adição de camadas bidirecionais conferiu aos modelos maior robustez na identificação de relações semânticas. Ao integrar informações contextuais anteriores e posteriores da sequência, essas camadas promoveram uma melhora significativa no desempenho global das predições. Paralelamente, a adição do mecanismo de atenção permitiu o foco nos segmentos mais relevantes das sentenças, resultando em um aumento na detecção de casos positivos.

A análise qualitativa das matrizes de confusão revelou padrões consistentes de erro, indicando uma forte sobreposição semântica entre as classes *other cyberbullying* e *not cyberbullying*. Esta limitação aponta para a necessidade de futuras investigações utilizando técnicas semi-supervisionadas ou não supervisionadas para melhorar a diferenciação de características das classes.

Ao contrário dos modelos baseados em Transformers **[18, 19]**, que dominam o estado da arte mas demandam recursos computacionais massivos, a abordagem proposta oferece um equilíbrio entre desempenho e custo. A baixa complexidade computacional viabiliza a implantação prática dessas soluções em redes sociais reais, promovendo ambientes mais seguros com menor custo operacional.

Para futuras contribuições, a continuidade desta pesquisa, propõem-se as seguintes direções:

DOI: 10.25286/repa.v11i4.3973

- Otimização: Utilização de otimizadores automáticos (ex: Optuna) para a busca de hiperparâmetros ideais.
- Refinamento: Implementação de estratégias de *fine-tuning* nas arquiteturas recorrentes.
- Embeddings Contextuais: Adoção do Sentence-BERT (SBERT) para criar representações vetoriais que capturem o contexto global da sentença, superando as limitações de *embeddings* baseados apenas em palavras isoladas.

7 REFERÊNCIAS

[1] SASIKUMAR, K. *et al*. Unmasking cyberbullies on social media platforms using machine learning. *In*: INTERNATIONAL CONFERENCE ON COMPUTING COMMUNICATION AND NETWORKING TECHNOLOGIES (ICCCNT), 14., 2023, Delhi, Índia. **Anais** [...] IEEE: Delhi, índia, 2023.

[2] JANIESCH, C.; ZSCHECH, P.; HEINRICH, K. Machine learning and deep learning. **Electronic Markets**, v. 31, n. 3, p. 685–695, 2021. DOI: https://doi.org/10.1007/s12525-021-00475-2. Disponível em: https://link.springer.com/article/10.1007/s12525-021-00475-2. Acesso em: 20 nov. 2025

[3] CHUN, J. S. *et al*. An international systematic review of cyberbullying measurements. **Computers in Human Behavior**, v. 113, p. 106485, 2020. DOI: https://doi.org/10.1016/j.chb.2020.106485. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0747563220302375. Acesso em: 20 nov. 2025

[4] MIENYE, I. D.; SWART, T. G.; OBAIDO, G. Recurrent neural networks: A comprehensive review of architectures, variants, and applications. **Information**, v. 15, n. 9, p. 517, 2024. DOI: https://doi.org/10.3390/info15090517. Disponível em: https://www.mdpi.com/2078-2489/15/9/517. Acesso em: 22 nov. 2025

[5] RAJ, C. *et al*. Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques. **Electronics**, v. 10, n. 22, p. 2810, 2021. DOI: https://doi.org/10.3390/electronics10222810. Disponível em: https://www.mdpi.com/2079-9292/10/22/2810. Acesso em: 22 nov. 2025

[6] PROTOPAPAS, P.; GLICKMAN, M. **Cs109b data science 2 lecture 10:** Recurrent neural networks. Institute for Applied Computational Science: Boston, MA, USA, 2019.

[7] ORUH, J.; VIRIRI, S.; ADEGUN, A. Long short-term memory recurrent neural network for automatic speech recognition. **IEEE Access**, v. 10, p. 30069–30079, 2022. DOI: https://doi.org/10.1109/ACCESS.2022.3159339. Disponível em: https://ieeexplore.ieee.org/abstract/document/9733937. Acesso em: 30 nov. 2025

[8] HASAN, M. *et al*. Bullyspotter: Online bullying detection in bengali text using transform learning. *In*: INTERNATIONAL CONFERENCE ON RECENT PROGRESSES IN SCIENCE, ENGINEERING AND TECHNOLOGY (ICRPSET*),* 2022, Rajshahi, Bangladesh. **Anais** [...] IEEE: Rajshahi, Bangladesh, 2022.

[9] MIKOLOV, T. *et al*. **Efficient Estimation of Word Representations in Vector Space**. 2013. Disponível em: https://arxiv.org/pdf/1301.3781. Acesso em: 30 nov. 2025

[10] JOULIN, A. *et al*. **FastText.zip:** Compressing text classification models. 2016. Disponível em: https://arxiv.org/abs/1612.03651. Acesso em: 01 dez. 2025

[11] JOULIN, A. *et al*. Bag of tricks for efficient text classification. *In:* PROCEEDINGS OF THE 15TH CONFERENCE OF THE EUROPEAN CHAPTER OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, 2., 2017, Valencia, Espanha. **Anais** [...] Short Papers: Valencia, Espanha, 2017.

[12] CHAPMAN, P. *et al*. **CRISP-DM 1.0:** Step-by-step data mining guide. [S.I.]: SPSS, 2000.

[13] OBAID, M. H.; GUIRGUIS, S. K.; ELKAFFAS, S. M. Cyberbullying detection and severity determination model. **Institute of Electrical and Eletronics Engineers**, v. 11, p. 97391–97399, 2023. DOI: https://doi.org/10.1109/ACCESS.2023.3313113.

Disponível em: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10243022. Acesso em: 03 dez. 2025

[14] HASAN, M. *et al*. Bullyspotter: Online bullying detection in bengali text using transform learning. *In:* INTERNATIONAL CONFERENCE ON RECENT PROGRESSES IN SCIENCE, ENGINEERING AND TECHNOLOGY (ICRPSET), 2022, Rajshahi, Bangladesh. **Anais** [...] IEEE: Rajshahi, Bangladesh, 2022.

[15] VASWANI, A. *et al*. Attention is all you need. Advances in neural information processing systems. *In:* CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 31., 2017, Long Beach, Califórnia, EUA. **Anais** [...] NIPS: Long Beach, Califórnia, EUA, 2017.

[16] GRAVES, A.; SCHMIDHUBER, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. **Neural networks**, v. 18, n. 5-6, p. 602-610, 2005. DOI: https://doi.org/10.1016/j.neunet.2005.06.042. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0893608005001206. Acesso em: 03 dez. 2025

[17] GERON, A. **Mãos à obra:** aprendizado de máquina com Scikit-Learn, Keras & Tensorflow. [S.l.]: Alta Books, 2021.

[18] GONGANE, V. U.; MUNOT, M. V.; ANUSE, A. Explainable AI for reliable detection of cyberbullying. *In*: IEEE PUNE SECTION INTERNATIONAL CONFERENCE (PUNECON), 2023, Pune, Índia. **Anais** [...] IEEE: Pune, Índia, 2023.

[19] GUPTA, S.; VADGAMA, U.; VEDHAVATHY, T. Identification and labeling of textual cyberbullying using bilstm and bert. *In*: INTERNATIONAL CONFERENCE ON NETWORKING AND COMMUNICATIONS (ICNWC), 2023, Chennai, índia. **Anais** [...] IEEE: Chennai, Índia, 2023

DOI: 10.25286/repa.v11i4.3973