Curupira: Arquiteturas Híbridas Transformer-BiLSTM para Detecção de Discurso Ofensivo em Português Brasileiro

Gabrielly Alves Gomes[1], **Iago de Sousa Aragão**[1]
Patricia Medyna Lauritzen de Lucena Drumond[1]

[1]Universidade Federal do Piauí (UFPI)
Campus Universitário Ministro Petrônio Portella - Ininga, Teresina - PI, 64049-550, Brasil

{gabrielly.gomes,iago.aragao,patriciamedyna}@ufpi.edu.br

Abstract. *This work presents Curupira, a controlled empirical study of neural architectures for offensive speech detection in Brazilian Portuguese under a rigorous methodological protocol: honest train/validation/test split, five random seeds, explicit parameter-capacity control, and paired bootstrap significance testing. Eight configurations are evaluated on OLID-BR, spanning four orders of magnitude in parameters. The results should be interpreted for this dataset and training regime, and indicate five main findings: (i) performance clusters around Macro-F1 close to 0.70, with BERTimbau (110M parameters) surpassing a lightweight Transformer→BiLSTM (350K) by 4.3 points; (ii) global attention provides statistically significant gains over a capacity-matched BiLSTM; (iii) increasing neural capacity does not improve performance in the tested setting; (iv) composition order matters, with Transformer→BiLSTM surpassing BiLSTM→Transformer by 6.3 points; (v) sinusoidal positional encoding over frozen embeddings substantially degrades performance, reducing Macro-F1 from 0.66 to 0.26.*

Resumo. *Este trabalho apresenta o Curupira, estudo empírico controlado de arquiteturas neurais para detecção de discurso ofensivo em português brasileiro sob protocolo metodológico rigoroso:* split *treino/validação/teste honesto, cinco sementes aleatórias, controle explícito de capacidade paramétrica e teste de significância via* bootstrap *pareado. Oito configurações são avaliadas no OLID-BR, cobrindo quatro ordens de magnitude em parâmetros. Os resultados devem ser interpretados para este conjunto de dados e regime de treinamento, e indicam cinco achados principais: (i) o desempenho se concentra próximo de Macro-F1 $=$ 0,70, com o BERTimbau (110M de parâmetros) superando um híbrido Transformer→BiLSTM leve (350K) em 4,3 pontos; (ii) a atenção global produz ganho estatisticamente significativo sobre uma BiLSTM equiparada em capacidade; (iii) aumentar a capacidade neural não melhora o desempenho no cenário avaliado; (iv) a ordem de composição importa, com Transformer→BiLSTM superando BiLSTM→Transformer em 6,3 pontos; (v) codificação posicional senoidal sobre* embeddings *congelados degrada substancialmente o desempenho, reduzindo o Macro-F1 de 0,66 para 0,26.*

1. Introdução

A proliferação de conteúdo ofensivo em plataformas digitais constitui problema relevante para comunidades *online* e para a sociedade, o que torna a moderação automatizada baseada em aprendizado de máquina uma estratégia necessária diante do volume de

publicações em redes sociais. Estudos sobre linguagem ofensiva e discurso de ódio em redes sociais, incluindo Twitter/X, mostram tanto a viabilidade quanto a complexidade da tarefa: distinguir *hate speech* de linguagem meramente ofensiva exige sensibilidade contextual, especialmente diante de coloquialidade, ironia e diferentes alvos de ataque [2, 19]. Embora a pesquisa tenha avançado para a língua inglesa, com recursos como o OLID [22] e as *shared tasks* OffensEval, o cenário para o português brasileiro permanece comparativamente menos explorado, conforme documentam estudos sobre ToLD-Br [12], HateBR [20] e anotação hierárquica de discurso de ódio em português [8].

Três lacunas convergentes motivam este trabalho. A primeira é a avaliação de arquiteturas híbridas Transformer-BiLSTM sob condições de baixo recurso, com *embeddings* estáticos, conjuntos de dados pequenos e ausência de pré-treino, condições frequentes em aplicações reais para PT-BR. A segunda é que trabalhos anteriores nem sempre controlam capacidade paramétrica, o que pode confundir efeito arquitetural com efeito de escala, problema sinalizado por Ramos et al. [16]. A terceira é que práticas comuns como uso de semente única e ausência de testes pareados comprometem a robustez das conclusões; Reimers et al. [17] demonstraram que a variância entre sementes pode inverter conclusões, e Dror et al. [5] argumentam que testes pareados devem constituir padrão mínimo.

Este trabalho apresenta o projeto Curupira[1], cujas contribuições podem ser sintetizadas em cinco eixos. A primeira contribuição (C1) é caracterizar, no OLID-BR e sob o protocolo avaliado, um agrupamento de desempenho próximo de Macro-F1 $= 0{,}70$ em oito modelos que cobrem quatro ordens de magnitude em parâmetros. A segunda (C2) é mostrar que o híbrido Transformer$\rightarrow$BiLSTM supera uma BiLSTM equiparada em 350K parâmetros com significância estatística via *bootstrap* pareado. A terceira (C3) é documentar que aumentar a capacidade dos modelos neurais leves não produziu ganho significativo neste cenário. A quarta (C4) é uma ablação de codificação posicional que mostra forte degradação quando PE senoidal é somada a *embeddings* congelados. A quinta (C5) é formalizar um protocolo metodológico replicável, com separação validação/teste, múltiplas sementes e comparação pareada.

2. Trabalhos Relacionados

2.1. Detecção de Discurso Ofensivo e Desbalanceamento

A detecção automática de discurso ofensivo evoluiu de métodos baseados em *features* manuais, como TF-IDF e n-gramas combinados com SVM ou regressão logística, para modelos de aprendizado profundo, conforme sintetizado por Fortuna e Nunes [7]. Estudos recentes indicam o predomínio de modelos baseados em *Transformers*, embora CNNs, LSTMs e arquiteturas híbridas ainda permaneçam competitivos em cenários com conjuntos de dados menores [13, 16]. Outro aspecto central é o desbalanceamento de classes: quando não tratado, a acurácia global tende a mascarar o desempenho da classe minoritária, o que justifica a adoção de Macro-F1 como métrica principal [12, 22].

2.2. BiLSTM, Transformer e Arquiteturas Híbridas

BiLSTMs e *Transformers* representam duas estratégias complementares para modelagem textual: a primeira enfatiza dependências sequenciais bidirecionais, enquanto a segunda

[1] *Curupira*: figura do folclore brasileiro, protetor das florestas, cujos pés voltados para trás confundem invasores. O nome evoca a função do modelo, reconhecer ataques e proteger o espaço digital.

utiliza *self-attention* para capturar relações globais entre *tokens* [9, 21]. Trabalhos anteriores exploraram a combinação dessas famílias em arquiteturas híbridas. Huang et al. [10], com o TRANS-BLSTM, e Khan et al. [11], com o BiCHAT, reportam ganhos em tarefas de entendimento de linguagem e detecção de discurso de ódio. Ainda assim, esses resultados costumam ser obtidos em condições mais favoráveis, com pré-treino massivo, *embeddings* treináveis ou maior escala de dados, e frequentemente sem controle explícito de capacidade paramétrica [16].

2.3. Português Brasileiro e Rigor Experimental

Os recursos para detecção de discurso ofensivo em português incluem o OLID-BR, adaptação do esquema OLID [23], o ToLD-Br apresentado por Leite et al. [12] com aproximadamente 21 mil *tweets*, o HateBR disponibilizado por Vargas et al. [20], com anotação especializada sobre comentários do Instagram, e o conjunto hierárquico de Fortuna et al. [8]. Estudos recentes também analisam repositórios de toxicidade em português [14] e o uso de LLMs na detecção de discurso de ódio em português [15]. O trabalho de Leite et al. [12] é particularmente relevante ao estabelecer que *transfer learning* do inglês não supera modelos monolíngues, que *zero-shot learning* alcança apenas Macro-F1 de 0,56 e que pelo menos seis mil exemplos são necessários para resultados confiáveis em PT-BR. O BERTimbau, disponibilizado por Souza et al. [18], estabeleceu-se como referência para a língua, com Macro-F1 de 0,76 no ToLD-Br em classificação binária.

Quanto ao rigor experimental, Reimers et al. [17] documentaram que resultados reportados com única semente não constituem evidência confiável, visto que a variância entre inicializações pode ser maior do que as diferenças supostas entre modelos. Baseados nessa evidência, Dror et al. [5] argumentam que a adoção de testes pareados deve ser padrão em NLP, e recomendam explicitamente *bootstrap* pareado para situações com poucas observações, nas quais o teste de Wilcoxon *signed-rank* sofre de baixa potência estatística. O *framework* sistemático para comparação de múltiplos classificadores oferecido por Demšar [3] e os métodos de *bootstrap* não paramétrico formalizados por Efron e Tibshirani [6] fornecem a base teórica do protocolo adotado na Seção 3.

3. Metodologia

3.1. Conjunto de Dados OLID-BR

O OLID-BR contém 6.472 *tweets* em português brasileiro, anotados para detecção binária de conteúdo ofensivo (OFF versus NOT). A análise exploratória conduzida na fase preliminar deste trabalho identificou 354 amostras rotuladas como OFF, porém sem *toxic_spans* válidos. Os *toxic_spans* são marcações de trechos específicos do texto identificados pelos anotadores como ofensivos, exigidas pelo esquema original do OLID-BR para todos os exemplos rotulados como OFF; sua ausência em amostras OFF configura inconsistência formal de anotação. Essas 354 amostras foram removidas do conjunto de treino, com base exclusivamente em propriedades dos metadados e antes de qualquer experimento arquitetural, o que exclui *data leakage* retroativo. O conjunto de teste foi preservado integralmente, de modo a refletir a distribuição original. Um experimento de sensibilidade conduzido sobre os três modelos centrais (BiLSTM-64, BiLSTM-128 e T$\rightarrow$B-*small*, cinco sementes cada) mostrou que a filtragem produz diferenças de Macro-F1 médio inferiores a $0{,}02$ pontos, sem alterar o ranking arquitetural ou as conclusões comparativas reportadas; a contribuição mais relevante da filtragem é a redução

da variância entre sementes, particularmente para o BiLSTM-128 (σ cai de $0{,}076$ para $0{,}032$). A Tabela 1 resume a distribuição resultante.

A diferença marginal de $-0{,}002$ pontos observada para o T→B-*small* situa-se dentro do erro de medição entre execuções, e o ranking arquitetural mantém-se inalterado em ambos os cenários.

O pré-processamento aplicado segue convenção comum na literatura para texto de redes sociais: conversão para minúsculas, remoção de URLs, menções de usuário, emojis, caracteres não alfabéticos (com preservação de diacríticos portugueses) e padrões de risadas frequentes (por exemplo, sequências como `kkk`, `haha` e `rsrs`). A tokenização adota vocabulário de 50 mil palavras e comprimento máximo de 100 *tokens*.

Tabela 1. Distribuição do OLID-BR antes e após filtragem de amostras OFF sem *toxic_spans* válidos.

	Sem filtragem			Com filtragem		
Conjunto	Total	OFF (%)	NOT (%)	Total	OFF (%)	NOT (%)
Treino	4.860	84,3	15,7	4.506	83,1	16,9
Validação (10%)	486	84,3	15,7	451	83,1	16,9
Teste	1.732	85,5	14,5	1.732	85,5	14,5

3.2. Protocolo de *Split* e Seleção de Modelo

Uma decisão metodológica central deste trabalho é a separação estrita entre conjuntos de validação e teste. Dez por cento do conjunto de treino foram reservados como conjunto de validação, estratificado por rótulo com semente fixa para garantir reprodutibilidade da partição. Todos os *callbacks* de seleção de modelo (`EarlyStopping`, `ModelCheckpoint` e `ReduceLROnPlateau`) operam exclusivamente sobre a perda de validação; o conjunto de teste é tocado uma única vez por execução, ao final do treinamento, apenas para reportar a métrica final. Essa disciplina evita a contaminação sutil, porém significativa, decorrente de selecionar a época ótima com base em desempenho no teste, conforme discutido por Dror et al. [5].

3.3. *Embeddings* e Ponderação de Classes

Os *embeddings* FastText pré-treinados em CommonCrawl para o português, disponibilizados por Bojanowski et al. (2017) com $d = 300$, são utilizados como camada de entrada dos modelos neurais leves e mantidos congelados (`trainable=False`) durante todo o treinamento. Essa decisão isola o efeito da composição arquitetural, impedindo que ganhos aparentes derivem de adaptação dos próprios *embeddings*. O desbalanceamento de classes é tratado por ponderação inversa à frequência, seguindo a fórmula $w_c = N/(k \cdot n_c)$, onde N é o total de amostras, k o número de classes e n_c a contagem da classe c. Sobre o treino filtrado, os pesos resultantes são $w_{\text{NOT}} = 2{,}96$ e $w_{\text{OFF}} = 0{,}60$.

3.4. Arquiteturas Avaliadas

Oito configurações foram avaliadas, organizadas em cinco famílias.

Baselines TF-IDF. Dois classificadores lineares foram treinados sobre vetores TF-IDF com n-gramas de tamanho um e dois, limitados a cinquenta mil *features*, o

primeiro baseado em regressão logística e o segundo em SVM linear, ambos com `class_weight='balanced'`. A função dessas referências é fornecer piso informativo para a comparação arquitetural posterior, respondendo à recomendação de Fortuna e Nunes [7] sobre o valor residual de *baselines* simples em regimes de baixo recurso.

BiLSTM puro. Duas configurações foram avaliadas, com sessenta e quatro (*BiLSTM-64*) e cento e vinte e oito unidades (*BiLSTM-128*). Cada variante recebe a sequência de *embeddings* FastText congelados, aplica uma camada BiLSTM com `return_sequences=True`, seguida de `GlobalMaxPool1D`, `Dense`(32, ReLU), `Dropout`(0,5) e `Dense`(1, sigmoide). As variantes totalizam aproximadamente 200K e 350K parâmetros, respectivamente, isolando o efeito da capacidade recorrente pura.

Híbrido Transformer→BiLSTM (T→B). A variante inspirada no TRANS-BLSTM [10] posiciona um bloco atencional antes da camada recorrente, na seguinte ordem: *embedding* congelado, Pre-LayerNorm, *MultiHeadAttention* com quatro cabeças e $d_k = 64$ com conexão residual, BiLSTM com `return_sequences=True` e `GlobalMaxPool1D`. Duas versões compartilham essa estrutura, diferenciando-se apenas no tamanho da camada recorrente: a *small* (BiLSTM de 64 unidades, aproximadamente 350K parâmetros) e a *large* (128 unidades, aproximadamente 500K). Nenhuma das duas inclui codificação posicional por padrão; o impacto dessa decisão é avaliado como ablação específica na Seção 4.

Híbrido BiLSTM→Transformer (B→T). A variante com ordem invertida aplica primeiro a BiLSTM (sessenta e quatro unidades, `return_sequences=True`) e em seguida o mesmo bloco atencional, totalizando aproximadamente 350K parâmetros. A comparação direta com a T→B-*small*, equiparada em capacidade paramétrica, isola o efeito da ordem de composição.

BERTimbau. O modelo `neuralmind/bert-base-portuguese-cased`, disponibilizado por Souza et al. [18], foi submetido a *fine-tuning* completo durante três épocas, com *batch* de oito, taxa de aprendizagem de 2×10^{-5}, precisão mista (`fp16`) e comprimento máximo de 128 *tokens*. Para manter consistência de protocolo com os modelos Keras, implementou-se um `WeightedLossTrainer` customizado sobre o `Trainer` do HuggingFace, aplicando *cross-entropy* ponderada pelos mesmos pesos de classe definidos para os modelos leves.

3.5. Configuração Experimental e Análise Estatística

Todos os modelos neurais foram treinados com otimizador Adam, perda *binary cross-entropy*, *batch size* 32 e máximo de dez épocas, com cinco sementes aleatórias por configuração no conjunto $\{42, 123, 456, 789, 1024\}$. Os experimentos foram executados em uma GPU NVIDIA A100 através do Google Colab, utilizando TensorFlow/Keras para os modelos leves e HuggingFace *Transformers* para o BERTimbau.

Para avaliação de significância estatística entre pares de modelos, adotou-se *bootstrap* pareado não paramétrico, conforme recomendado por Dror et al. [5] para cenários com número limitado de observações. Dadas as cinco medições pareadas por semente, o vetor de diferenças foi reamostrado com reposição 10 mil vezes, computando-se o intervalo de confiança de 95% da média da diferença. Uma diferença foi considerada estatisticamente significativa quando o intervalo de confiança não contém zero. A escolha do *bootstrap* em vez do Wilcoxon *signed-rank* se justifica pela observação de que o valor-p

mínimo alcançável pelo Wilcoxon com $n = 5$ é $0{,}0625$, tornando o teste incapaz de rejeitar a hipótese nula mesmo na presença de diferenças reais, problema discutido por Dror et al. [5] e compatível com as recomendações de Efron e Tibshirani [6] para inferência em amostras pequenas. O desvio-padrão σ de Macro-F1 entre sementes é reportado com intervalo de confiança de 95% obtido por *bootstrap* sobre a amostra das cinco sementes.

4. Resultados

4.1. Ranking Final e Significância Estatística

A Tabela 2 consolida o desempenho das oito configurações no conjunto de teste, ordenadas por Macro-F1 médio.

Tabela 2. Ranking final no conjunto de teste (média $\pm\,\sigma$ sobre 5 sementes).

#	Modelo	Macro-F1	NOT-F1	OFF-F1	Parâm.
1	BERTimbau	$\mathbf{0{,}705 \pm 0{,}012}$	$\mathbf{0{,}508 \pm 0{,}008}$	$\mathbf{0{,}901 \pm 0{,}022}$	$\sim$110M
2	Híbrido T$\rightarrow$B-*small*	$0{,}662 \pm 0{,}008$	$0{,}478 \pm 0{,}007$	$0{,}844 \pm 0{,}014$	$\sim$350K
3	Híbrido T$\rightarrow$B-*large*	$0{,}656 \pm 0{,}013$	$0{,}469 \pm 0{,}016$	$0{,}834 \pm 0{,}028$	$\sim$500K
4	BiLSTM-64	$0{,}630 \pm 0{,}017$	$0{,}440 \pm 0{,}011$	$0{,}820 \pm 0{,}023$	$\sim$200K
5	TF-IDF + LR	0,619	0,365	0,873	$\sim$50K
6	BiLSTM-128	$0{,}613 \pm 0{,}032$	$0{,}422 \pm 0{,}031$	$0{,}804 \pm 0{,}039$	$\sim$350K
7	TF-IDF + SVM	0,600	0,297	0,903	$\sim$50K
8	Híbrido B$\rightarrow$T-*small*	$0{,}598 \pm 0{,}044$	$0{,}405 \pm 0{,}052$	$0{,}791 \pm 0{,}031$	$\sim$350K

Vale destacar um padrão interessante nas métricas por classe: o TF-IDF+SVM apresenta o maior OFF-F1 de todos os modelos avaliados (0,903), superando inclusive o BERTimbau (0,901). O ganho vem, contudo, ao custo do menor NOT-F1 do conjunto (0,297), indicando estratégia de classificação enviesada para a classe majoritária. Essa observação valida empiricamente a adoção de Macro-F1 como métrica principal: otimização da classe majoritária isoladamente pode produzir modelos enganosamente competitivos em métricas agregadas como acurácia, mascarando desempenho insatisfatório na classe minoritária.

O BERTimbau lidera o ranking com $0{,}705 \pm 0{,}012$, seguido pelo híbrido T$\rightarrow$B-*small* com $0{,}662 \pm 0{,}008$. A configuração B$\rightarrow$T-*small* ocupa a última posição entre os modelos neurais, com $0{,}598 \pm 0{,}044$, abaixo inclusive dos *baselines* TF-IDF. A amplitude total entre o melhor e o pior modelo é de aproximadamente 10,7 pontos percentuais, apesar da cobertura de quatro ordens de magnitude em número de parâmetros, de 50 mil (TF-IDF) a 110 milhões (BERTimbau).

A Tabela 3 apresenta os testes de significância via *bootstrap* pareado, conforme o protocolo recomendado por Dror et al. [5]. Os resultados são discutidos em detalhe na Seção 5.

4.2. Métricas por Classe

A classe minoritária NOT constitui gargalo universal. Mesmo o BERTimbau atinge apenas NOT-F1 $= 0{,}508$, e todos os modelos exibem NOT-F1 entre 0,30 e 0,51. O

Tabela 3. *Bootstrap* pareado das diferenças de Macro-F1 (A − B), 10 mil reamostragens.

A	B	Diferença	IC 95%	Sig. ($\alpha = 0{,}05$)
BERTimbau	T→B-*small*	$+0{,}043$	$[+0{,}033;\ +0{,}053]$	✓
BERTimbau	BiLSTM-128	$+0{,}092$	$[+0{,}071;\ +0{,}114]$	✓
T→B-*small*	BiLSTM-128	$+0{,}049$	$[+0{,}027;\ +0{,}071]$	✓
T→B-*small*	B→T-*small*	$+0{,}063$	$[+0{,}030;\ +0{,}097]$	✓
T→B-*large*	BiLSTM-128	$+0{,}040$	$[+0{,}015;\ +0{,}062]$	✓
T→B-*small*	T→B-*large*	$+0{,}009$	$[-0{,}002;\ +0{,}020]$	n.s.
BiLSTM-128	BiLSTM-64	$-0{,}017$	$[-0{,}048;\ +0{,}007]$	n.s.

T→B-*small* obtém NOT-F1 de $0{,}478$, superior ao BiLSTM-128 ($0{,}422$) e ao BiLSTM-64 ($0{,}440$). O OFF-F1 é uniformemente alto entre os modelos neurais (acima de 0,79), refletindo o predomínio da classe majoritária no conjunto de teste.

4.3. Ablação: Codificação Posicional e *Layer Normalization*

A Tabela 4 resume a ablação de codificação posicional. A adição de PE senoidal reduz o Macro-F1 de $0{,}657 \pm 0{,}009$ para $0{,}260 \pm 0{,}183$, com colapso completo da classe minoritária em três de cinco sementes (NOT-F1 igual a zero). Trata-se do efeito arquitetural mais expressivo observado no trabalho, tanto em magnitude quanto em consistência da degradação.

Uma ablação complementar compara Pre-LN, adotada como padrão, e Post-LN sobre a mesma variante T→B-*small*. Os resultados são virtualmente indistinguíveis ($0{,}662 \pm 0{,}008$ contra $0{,}664 \pm 0{,}005$), com intervalo de confiança da diferença pareada $[-0{,}009;\ +0{,}004]$, indicando que a escolha de *LayerNorm* não altera o desempenho neste regime.

Tabela 4. Ablação de *positional encoding* sobre o T→B-*small* (5 sementes).

Configuração	Macro-F1	NOT-F1	OFF-F1	Acc.
Sem PE	$0{,}657 \pm 0{,}009$	$0{,}474 \pm 0{,}008$	$0{,}840 \pm 0{,}011$	$0{,}755 \pm 0{,}014$
Com PE senoidal	$0{,}260 \pm 0{,}183$	$0{,}101 \pm 0{,}140$	$0{,}420 \pm 0{,}506$	$0{,}429 \pm 0{,}395$

5. Discussão

5.1. Desempenho no OLID-BR e Custo-Benefício dos Modelos Leves

Os oito modelos avaliados, cobrindo quatro ordens de magnitude em parâmetros, ficaram em um intervalo relativamente estreito de Macro-F1, entre 0,60 e 0,71. O BERTimbau, com aproximadamente 110 milhões de parâmetros, supera o T→B-*small* (∼350 mil parâmetros) em 4,3 pontos percentuais. Esses resultados não demonstram um limite geral do OLID-BR, mas sugerem que, sob o protocolo experimental adotado e com cerca de 4,5 mil amostras de treino após a filtragem descrita na Seção 3, os ganhos por aumento de capacidade ou refinamento arquitetural são limitados. Essa leitura é compatível com Leite et al. [12], que identificaram seis mil exemplos como piso para resultados confiáveis

em PT-BR. Como implicação prática, a agregação entre OLID-BR, ToLD-Br e HateBR [12, 20] permanece uma direção importante para avaliar se o padrão observado se mantém em corpora maiores e mais diversos.

Esse padrão também tem implicação prática de custo-benefício. O classificador TF-IDF seguido de regressão logística atinge Macro-F1 de 0,619 em menos de dez segundos de treinamento, sem uso de GPU, o que corresponde a aproximadamente 87,8% do desempenho do BERTimbau utilizando cerca de 0,05% dos parâmetros. Em cenários com restrição computacional severa, incluindo moderação em tempo real, sistemas *edge* e *pipelines* de alto volume, a diferença de 8,6 pontos percentuais pode ser aceitável quando se consideram o custo marginal por inferência e os requisitos de latência.

5.2. A Atenção Importa

O achado mais relevante da Tabela 3 é a superioridade estatisticamente significativa do T→B-*small* sobre o BiLSTM-128, dado que ambos possuem a mesma ordem de parâmetros (∼350K). Em experimentos sem controle explícito de capacidade, seria natural atribuir o ganho dos híbridos ao número maior de parâmetros, interpretação contra a qual Ramos et al. [16] alertam em sua revisão ao discutir *ensembles* BiLSTM-*Transformer*. Os resultados aqui apresentados indicam que, neste regime experimental, o mecanismo de *self-attention* contribui para o desempenho ao modelar coocorrências globais entre *tokens* que a recorrência bidirecional local pode não capturar. Esse resultado dialoga, em escala de parâmetros muito menor, com a observação qualitativa de Khan et al. [11] sobre os ganhos da atenção hierárquica no BiCHAT. Ainda assim, trata-se de evidência condicionada ao OLID-BR, aos *embeddings* congelados e às arquiteturas avaliadas, não de uma conclusão universal sobre atenção global em classificação textual.

5.3. Saturação Paramétrica e Variância entre Sementes

Dobrar a capacidade do BiLSTM (de 64 para 128 unidades) não produziu ganho neste protocolo e, inclusive, associa-se a leve deterioração do desempenho médio (diferença de $-0{,}017$, IC 95% $[-0{,}048;\ +0{,}007]$). De modo análogo, o T→B-*large* (∼500K parâmetros) não supera o T→B-*small* (∼350K). Dois mecanismos concorrentes podem explicar esse padrão. O primeiro é saturação informacional local: em regime com poucos milhares de exemplos, a capacidade mínima necessária pode já ser satisfeita pelos modelos menores, de modo que parâmetros adicionais não recebem sinal útil suficiente. O segundo é *overfitting* ao ruído do conjunto de dados: capacidade adicional pode permitir ao modelo memorizar padrões espúrios do treino, produzindo deterioração líquida no teste, apesar da manutenção de *dropout*, *early stopping* e ponderação de classes. A combinação desses dois efeitos é uma hipótese plausível para a direção observada, mas sua confirmação exige avaliação em outros conjuntos de dados e regimes de treinamento.

Vale destacar que o BiLSTM-128 varia de 0,571 a 0,652 entre sementes, amplitude de oito pontos percentuais. Um relato com a semente 1024 concluiria que o BiLSTM-128 é competitivo; com a semente 456, concluiria que é deficiente. Essa vulnerabilidade, já documentada por Reimers et al. [17] e sistematizada por Dror et al. [5], reforça a necessidade de reportar resultados agregados sobre múltiplas sementes como prática metodológica padrão.

5.4. A Ordem de Composição Importa

O T→B-*small* supera o B→T-*small* em 6,3 pontos percentuais (IC 95% $[+0{,}030;\ +0{,}097]$) e apresenta variância substancialmente menor. Os resultados sugerem que, neste regime com *embeddings* FastText congelados, aplicar atenção antes da recorrência é mais adequado, possivelmente porque o bloco atencional opera sobre representações mais estáveis antes da variabilidade introduzida pela BiLSTM. Assim, além do tipo de componente utilizado, a ordem de composição mostra efeito relevante tanto em desempenho quanto em estabilidade.

5.5. Por que o *Positional Encoding* Degrada o Modelo

A adição de codificação posicional senoidal ao T→B-*small* reduz o Macro-F1 para $0{,}260$ e produz colapso da classe NOT em parte das sementes. Nesse regime, a combinação entre PE fixo e *embeddings* FastText congelados parece perturbar a geometria original das representações sem oferecer capacidade de adaptação durante o treinamento. Embora o mecanismo exato ainda demande ablação específica, o resultado indica que, neste cenário, atenção sem PE foi consistentemente superior. PE aprendível, escalonado ou aplicado sobre representações contextualizadas permanece como hipótese para trabalho futuro.

5.6. Análise Qualitativa de Erros

Para compreender as diferenças de comportamento entre modelos, foram analisadas 168 instâncias do conjunto de teste nas quais o T→B-*small* classificou corretamente e o BiLSTM-64 errou, considerando a semente 42: 150 textos OFF capturados exclusivamente pelo T→B e 18 textos NOT classificados corretamente apenas pelo T→B.

Entre os OFF capturados exclusivamente pelo T→B, predominam ofensas indiretas, contextuais ou irônicas, sem palavras-chave explicitamente ofensivas. Exemplos representativos incluem “essa *temp* me matou de vergonha alheia”, caracterizada por deboche implícito; “me senti uma idiota (e eu fui)”, marcada por autoironia com carga ofensiva; e “Simplesmente patético tudo isso”, que carrega juízo pejorativo sem xingamento direto. O mecanismo de *self-attention* pondera a coocorrência global de termos aparentemente neutros em posição discriminativa, enquanto a recorrência depende de padrões sequenciais locais e falha quando não há gatilhos lexicais explícitos.

Entre os NOT acertados apenas pelo T→B, observam-se textos com palavras potencialmente ofensivas utilizadas fora de contexto ofensivo. Exemplos incluem “nossa isso deu mó treta na época”, relato neutro sobre conflito passado; “Pobre diabo.”, expressão coloquial sem endereçamento agressivo; e “O insta da racista”, acusação legítima e não ofensa em si. O BiLSTM, dependente de padrões lexicais locais, dispara em palavras isoladas; o T→B, ao ponderar o contexto global, identifica que o tom geral não é ofensivo.

5.7. Limitações

A investigação cobriu um único conjunto de dados, o OLID-BR, o que deixa em aberto a generalização para corpora mais amplos como o ToLD-Br [12], o HateBR [20] ou repositórios agregados de toxicidade em português [14]. Cinco sementes aleatórias por configuração constituem o mínimo prático para *bootstrap* pareado; a ampliação para dez ou mais sementes produziria intervalos de confiança mais estreitos, particularmente úteis

para comparações próximas ao limiar de significância. O *fine-tuning* do BERTimbau utilizou hiperparâmetros padrão da literatura, sem busca em grade sistemática, o que pode subestimar o melhor desempenho possível desse modelo. A análise qualitativa foi conduzida sobre uma única semente, estratégia comum na literatura, mas que deixa margem para variação anedótica. Também não foram avaliados *embeddings* FastText treináveis, *encoders* contextualizados dentro da arquitetura híbrida, nem variantes aprendíveis ou escalonadas de codificação posicional. Por fim, a calibração do limiar de decisão com foco na classe minoritária permanece como hipótese de baixo custo para elevar NOT-F1 sem retreinamento dos modelos.

6. Conclusão e Trabalhos Futuros

Este trabalho apresentou o projeto Curupira, investigação empírica controlada de arquiteturas neurais para detecção de discurso ofensivo em português brasileiro, conduzida sob protocolo metodológico rigoroso, que incluiu separação estrita entre validação e teste, cinco sementes aleatórias por configuração, controle explícito de capacidade paramétrica e teste de significância via *bootstrap* pareado. Cinco achados principais emergem do conjunto experimental e devem ser interpretados no escopo do OLID-BR e do regime de treinamento adotado. Primeiro, o desempenho dos modelos avaliados se concentrou próximo de Macro-F1 0,70, com o BERTimbau superando o melhor modelo leve em 4,3 pontos percentuais. Segundo, a atenção global produziu ganho estatisticamente significativo sobre uma BiLSTM equiparada em capacidade. Terceiro, aumentar a capacidade dos modelos neurais leves não produziu ganho significativo neste cenário. Quarto, a ordem de composição importou, com a variante Transformer→BiLSTM superando a composição inversa em desempenho e estabilidade entre sementes. Quinto, a codificação posicional senoidal somada a *embeddings* congelados degradou substancialmente o desempenho, configurando achado útil para trabalhos futuros com *embeddings* estáticos. O contraste entre o que se observa sob protocolo rigoroso e o que seria obtido em configurações experimentais comuns reforça o chamado de Reimers et al. [17] e Dror et al. [5] pela adoção de múltiplas sementes e testes pareados como padrão em avaliação de modelos de NLP.

Como desdobramentos futuros, quatro direções são consideradas promissoras. A primeira é a agregação dos principais recursos para PT-BR (OLID-BR, ToLD-Br, HateBR e repositórios correlatos) em um corpus unificado, permitindo avaliar se o agrupamento de desempenho observado é específico do OLID-BR ou se persiste em escala maior. A segunda é a substituição do FastText pelo BERTimbau como *encoder* em arquitetura híbrida, combinada com ablação sistemática entre PE aprendível, escalonado e senoidal. A terceira é a calibração do limiar de decisão orientada à classe minoritária, estratégia de baixo custo com potencial de elevar NOT-F1 sem retreinamento. A quarta é a ampliação para dez ou mais sementes por configuração, refinando os intervalos de confiança das comparações fronteiriças. O código e os dados processados serão disponibilizados em repositório público.

Considerações Éticas

Este trabalho utiliza o OLID-BR, conjunto de dados previamente disponibilizado para fins de pesquisa, sem coleta adicional de dados, interação com participantes humanos ou tentativa de reidentificação de usuários. Como a tarefa envolve detecção de discurso ofensivo, reconhece-se que erros de classificação podem gerar impactos relevantes, incluindo

falsos positivos sobre expressões legítimas e falsos negativos sobre conteúdos ofensivos. Assim, os modelos avaliados devem ser interpretados como apoio à análise e não como mecanismo autônomo de decisão em sistemas reais de moderação, os quais exigiriam supervisão humana, auditoria e avaliação contextual de vieses.

Declaração de Uso de IA Generativa

Ferramentas de IA generativa foram utilizadas como apoio auxiliar na pesquisa e na escrita. No projeto Curupira, elas apoiaram decisões de implementação, depuração, melhoria de erros e orientação sobre aspectos de código, sem geração automática do código final; a execução, validação dos resultados e análise permaneceram sob responsabilidade humana. Na escrita, auxiliaram na estruturação do artigo conforme as seções solicitadas, na criação de tópicos-guia, na sugestão de título, resumo e *abstract*, e na adequação de trechos ao formato LaTeX. Todo o conteúdo foi revisado pelos autores para evitar desvios do escopo da pesquisa, inconsistências técnicas ou informações inventadas. As ferramentas utilizadas foram Gemini 3.1 Pro na fase inicial, Claude Code Opus 4.7 com pensamento adaptativo no apoio técnico final, e Claude Opus 4.6 no apoio à escrita.

Referências

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

[2] Davidson, T., Warmsley, D., Macy, M. W. e Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, p. 512–515.

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

[4] Devlin, J., Chang, M.-W., Lee, K. e Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, p. 4171–4186.

[5] Dror, R., Baumer, G., Shlomov, S. e Reichart, R. (2018). The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of ACL*, p. 1383–1392.

[6] Efron, B. e Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. CRC Press, Boca Raton.

[7] Fortuna, P. e Nunes, S. (2018). A survey on automatic detection of hate speech in text. *ACM Computing Surveys*, 51(4):1–30.

[8] Fortuna, P., Rocha da Silva, J., Soler-Company, J., Wanner, L. e Nunes, S. (2019). A hierarchically-labeled Portuguese hate speech dataset. In *Proceedings of the Third Workshop on Abusive Language Online*, p. 94–104.

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

[10] Huang, Z., Xu, P., Liang, D., Mishra, A. e Xiang, B. (2020). TRANS-BLSTM: Transformer with bidirectional LSTM for language understanding. *arXiv preprint arXiv:2003.07000*.

[11] Khan, S., Fazil, M., Sejwal, V. K., Alshara, M. A., Alotaibi, R. M., Kamal, A. e Baig, A. R. (2022). BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection. *Journal of King Saud University, Computer and Information Sciences*, 34(7):4335–4344.

[12] Leite, J. A., Silva, D. F., Bontcheva, K. e Aker, A. (2020). Toxic language detection in social media for Brazilian Portuguese: New dataset and multilingual analysis. In *Proceedings of AACL-IJCNLP*, p. 914–924.

[13] Malik, J. S., Pang, G. e van den Hengel, A. (2023). Deep learning for hate speech detection: A comparative study. *arXiv preprint arXiv:2202.09517*.

[14] Moreira, L. S., Gibrim, P. T. M., Rocha, L. e Reis, J. C. S. (2026). Anatomy of Data Repositories for the Analysis and Detection of Toxicity in Portuguese. In *Proceedings of the 17th International Conference on Computational Processing of Portuguese (PROPOR 2026) – Vol. 1*, p. 456–466.

[15] Oliveira, A., Cecote, T., Silva, P., Gertrudes, J., Freitas, V. e Luz, E. (2023). How Good Is ChatGPT For Detecting Hate Speech In Portuguese? In *Proceedings of the 14th Brazilian Symposium in Information and Human Language Technology*, p. 103–112.

[16] Ramos, G., Batista, F., Ribeiro, R., Fialho, P., Moro, S., Fonseca, A., Guerra, R., Carvalho, P., Marques, C. e Silva, C. (2024). A comprehensive review on automatic hate speech detection in the age of the transformer. *Social Network Analysis and Mining*, 14:204.

[17] Reimers, N. e Gurevych, I. (2017). Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proceedings of EMNLP*, p. 338–348.

[18] Souza, F., Nogueira, R. e Lotufo, R. (2020). BERTimbau: Pretrained BERT models for Brazilian Portuguese. In *Proceedings of BRACIS*, p. 403–417.

[19] Silva, L. A., Mondal, M., Correa, D., Benevenuto, F. e Weber, I. (2016). Analyzing the targets of hate in online social media. In *Proceedings of the Tenth International AAAI Conference on Web and Social Media*, p. 687–690.

[20] Vargas, F. A., Carvalho, I., Rodrigues de Góes, F., Pardo, T. A. S. e Benevenuto, F. (2022). HateBR: A large expert annotated corpus of Brazilian Instagram comments for offensive language and hate speech detection. In *Proceedings of LREC*, p. 7174–7183.

[21] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. e Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems 30*, p. 5998–6008.

[22] Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N. e Kumar, R. (2019). Predicting the type and target of offensive posts in social media. In *Proceedings of NAACL-HLT*, p. 1415–1420.

[23] Trajano, Douglas, Bordini, Rafael H. e Vieira, Renata (2024). OLID-BR: offensive language identification dataset for Brazilian Portuguese. *Language Resources and Evaluation*, 58(4):1263–1289.