Análise Comparativa entre Modelos BiLSTM e BERT para Classificação de Sentimentos em Tweets

1st Amanda Moura
Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
amanda.souza.1381861@sga.pucminas.br

2nd André Faria
Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
andre.faria923006@sga.pucminas.br

3rd Luiz Gustavo Santos
Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
luiz.mendes@sga.pucminas.br

4th Pedro Ramos
Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
pedro.vidigal@sga.pucminas.br

5th Philippe Vieira
Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
philippe.vieira@sga.pucminas.br

***Abstract*—Este trabalho apresenta uma análise comparativa entre dois modelos amplamente utilizados em Processamento de Linguagem Natural: BiLSTM (Long Short-Term Memory Bidirecional) e BERT (Bidirectional Encoder Representations from Transformers). A pesquisa investiga a classificação de sentimentos em mensagens oriundas de redes sociais, particularmente tweets. Os modelos foram implementados utilizando Python no ambiente Google Colab. O framework Keras (parte do TensorFlow) foi utilizado para a implementação do BiLSTM e o BERT foi construído utilizando o framework PyTorch. A preparação dos dados foi realizada utilizando bibliotecas Python e ferramentas de pré-processamento do Keras e do Hugging Face. A avaliação do desempenho dos modelos foi conduzida utilizando métricas padrão para tarefas de classificação, como Acurácia, Precisão, Recall e F1-Score.**

***Index Terms*—Classificação de Sentimentos, LSTM, BERT, Transformers, Processamento de Linguagem Natural, Tweets.**

I. Introdução

Com o crescimento exponencial das redes sociais e plataformas digitais, tornou-se essencial o desenvolvimento de ferramentas computacionais que possam interpretar, classificar e extrair valor dos dados gerados diariamente por usuários ao redor do mundo. Um dos principais campos dentro dessa área é a análise de sentimentos, que visa identificar a polaridade (positiva, negativa ou neutra) de uma opinião expressa em um texto. Essa técnica tem aplicações diretas em áreas como marketing, política, gestão de marcas e análise de comportamento de usuários.

O principal desafio da análise de sentimentos está no tipo de linguagem encontrada em redes sociais. Diferente de textos mais estruturados e formais, como artigos de notícias ou documentos acadêmicos, mensagens em plataformas como o Twitter tendem a ser curtas, ambíguas, com uso de ironia, emojis, abreviações e um vocabulário altamente dinâmico. Esses elementos tornam a tarefa de análise automática de sentimentos muito mais complexa.

Tradicionalmente, modelos baseados em redes neurais recorrentes, como o LSTM (Long Short-Term Memory), têm se mostrado eficazes para o processamento de sequências textuais. Sua capacidade de capturar dependências temporais e relacionamentos entre palavras em uma sequência faz com que sejam amplamente utilizados em tarefas de PLN (Processamento de Linguagem Natural). No entanto, nos últimos anos, uma nova classe de modelos baseada em arquiteturas Transformer, especialmente com o advento do BERT, tem se destacado pela capacidade de capturar relações contextuais de forma mais eficaz, processando o texto em paralelo e de maneira bidirecional.

Diante desse cenário, surge um problema relevante na literatura: embora existam estudos independentes sobre o desempenho de LSTM e Transformers, ainda há uma carência de investigações comparativas sistemáticas que analisem o comportamento dessas duas abordagens em contextos informais como o das redes sociais. Esta lacuna torna-se ainda mais significativa quando se busca orientar pesquisadores e profissionais sobre qual tecnologia é mais adequada para análise de sentimentos em ambientes digitais.

Assim, o objetivo deste trabalho é analisar comparativamente o modelo BiLSTM e o modelo BERT-base-uncased, no processamento de linguagem natural para classificação de sentimento em tweets, utilizando como base de dados um

conjunto de tweets rotulados disponível na plataforma Kaggle. A avaliação e comparação do desempenho dos modelos é feita com base em métricas comuns para classificação de texto, Acurácia, Precisão, Recall e F1-Score para cada classe de sentimento. Essas métricas fornecem uma visão detalhada da capacidade dos modelos em classificar os tweets nos diferentes rótulos de sentimento (Positivo, Negativo, Neutro, Irrelevante), permitindo uma comparação robusta entre as abordagens BiLSTM e BERT. Justifica-se este estudo pela necessidade crescente de compreender os limites e as vantagens de cada abordagem em contextos reais e desafiadores, fornecendo subsídios para a escolha mais adequada de modelos em aplicações práticas.

II. Método

Esta seção descreve a metodologia adotada para a análise comparativa entre os modelos BiLSTM e BERT na tarefa de classificação de sentimentos em tweets. O estudo utilizou um conjunto de dados composto por tweets rotulados, organizados em dois arquivos CSV, um para treinamento e outro para validação. Cada registro no conjunto de dados contém quatro campos: TweetID (identificador único do tweet), Entity (entidade mencionada no tweet, como nome de empresa, produto ou pessoa), Sentiment (classificação do sentimento) e TweetContent (conteúdo textual do tweet).

A. Tratamento de dados

O pré-processamento dos dados foi adaptado às necessidades específicas de cada modelo, respeitando suas características arquiteturais. Para ambos os modelos, é feita uma etapa inicial de limpeza dos dados, removendo registros com valores ausentes nas colunas de conteúdo e relatório de sentimento, a distinção de maiúsculas e minúsculas é padronizada, URLs removidas e espaços em branco normalizados. A partir desta limpeza básica, foram adotadas estratégias específicas para cada modelo.

Para o BiLSTM, é aplicada uma limpeza mais rigorosa, removendo caracteres não alfabéticos e menções de usuário, a entidade e o tweet são concatenados com um token separador para contextualização, uma tokenização utilizando o tokenizador do Keras é realizada, os textos convertidos para sequências numéricas, é aplicado um padding para uniformizar o comprimento das sequências em 150 tokens, os rótulos convertidos para formato *one-hot encoding*, e o vocabulário é limitado aos 20.000 termos mais frequentes.

Para o BERT, uma abordagem menos invasiva de pré-processamento foi escolhida, utilizando o tokenizador específico do BERT (bert-base-uncased), codificando a entidade e o tweet como um par de sentenças, e mantendo o comprimento máximo da sequência limitado em 128 tokens.

A diferenciação no pré-processamento reflete as características de cada arquitetura. O BiLSTM, sendo um modelo mais simples, beneficia-se de uma limpeza mais rigorosa que reduza ruídos e simplifique o texto. Já o BERT, com sua arquitetura baseada em Transformer e tokenização por subpalavras (WordPiece), é mais robusto a variações linguísticas e capaz de lidar com palavras desconhecidas, justificando uma abordagem menos invasiva que preserva mais características originais do texto.

B. Arquitetura dos Modelos

O modelo BiLSTM é implementado utilizando a API Keras do TensorFlow, com a seguinte arquitetura em camadas:

Camada de Embedding (dimensão 128): Transforma os índices de palavras em vetores densos.

SpatialDropout1D (taxa 0.3): Aplica regularização na camada de embedding, desativando canais inteiros.

Primeira camada BiLSTM (128 unidades): Processa a sequência em ambas as direções com dropout recorrente de 0.3.

Segunda camada BiLSTM (64 unidades): Captura padrões de mais alto nível com dropout recorrente de 0.2.

Camada Densa (64 unidades): Integra as características extraídas com ativação ReLU.

Dropout (taxa 0.5): Adiciona regularização para prevenir overfitting.

Camada de saída (4 unidades): Produz probabilidades para cada classe usando ativação softmax.

A arquitetura em duas camadas é utilizada para capturar tanto características locais quanto dependências de longo alcance no texto. O uso extensivo de *dropout* em diferentes níveis (*embedding*, recorrente e entre camadas densas) visa prevenir o *overfitting*, considerando especialmente o tamanho limitado do conjunto de dados. A dimensionalidade decrescente ($128 \rightarrow 64$) nas camadas BiLSTM segue o princípio de afunilamento, permitindo abstração progressiva das características.

Para o BERT, é utilizado o modelo pré-treinado *bert-base-uncased* da biblioteca Hugging Face Transformers, realizando *fine-tuning* para a tarefa específica:

Modelo base BERT: 12 camadas de codificador, 768 dimensões ocultas e 12 cabeças de atenção.

Pooling da representação [CLS]: Extração do token especial que captura a representação da sequência.

Camada de classificação: Adaptada para 4 classes de saída.

O modelo *base uncased* é utilizado por ser um equilíbrio entre capacidade representacional e eficiência computacional. E é mantida a arquitetura original do Transformer, modificando somente a camada final de classificação, aproveitando assim o conhecimento linguístico adquirido durante o pré-treinamento.

C. Treinamento dos modelos

O treinamento do modelo BiLSTM é configurado com os seguintes parâmetros:

- Otimizador: Adam
- Função de perda: Categorical Crossentropy
- Batch size: 128
- Épocas máximas: 10
- Early stopping com paciência de 3 épocas
- Redução da taxa de aprendizado em platôs

A implementação de callbacks como early stopping e redução da taxa de aprendizado visa otimizar o processo de

treinamento, evitando overfitting e facilitando a convergência do modelo.

Para o *fine-tuning* do BERT, foi adotada uma abordagem mais conservadora:

- Otimizador: AdamW com taxa de aprendizado de 2e-5
- Função de perda: Cross Entropy
- Batch size: 32 (menor devido à maior demanda de memória)
- Épocas: 3
- Scheduler de taxa de aprendizado linear com warmup
- Clipping de gradiente para estabilidade

O número reduzido de épocas para o BERT é uma prática comum em *fine-tuning* de modelos pré-treinados, pois estes já possuem conhecimento linguístico substancial e tendem a convergir mais rapidamente, além de serem mais suscetíveis ao overfitting com treinamento prolongado.

D. Avaliação dos Modelos

A avaliação dos modelos foi realizada utilizando métricas complementares para uma análise abrangente:

- **Precisão por classe:** Capacidade do modelo de evitar falsos positivos para cada sentimento.
- **Recall por classe:** Capacidade do modelo de identificar corretamente todos os exemplos de cada sentimento.
- **F1-Score:** Média harmônica entre precisão e recall, equilibrando ambas as métricas.
- **Matriz de Confusão:** Visualização detalhada da distribuição de erros entre classes.
- **Curvas de Aprendizado:** Análise da evolução da acurácia e perda durante o treinamento.

Todos os experimentos foram conduzidos no ambiente Google Colab, utilizando aceleradores GPU para otimizar o tempo de treinamento. As principais bibliotecas utilizadas foram: TensorFlow e Keras para implementação do BiLSTM; PyTorch e Hugging Face Transformers para implementação do BERT; Scikit-learn para métricas de avaliação; Pandas e NumPy para manipulação de dados; Matplotlib e Seaborn para visualizações.

A metodologia adotada visa explorar as diferenças fundamentais entre modelos recorrentes bidirecionais (BiLSTM) e modelos baseados em atenção (BERT) para a tarefa específica de classificação de sentimentos em tweets. A escolha de manter fluxos de pré-processamento distintos para cada modelo reflete a intenção de otimizar o desempenho individual de cada arquitetura, respeitando suas características intrínsecas.

A inclusão da entidade como parte do contexto de entrada para ambos os modelos representa uma contribuição metodológica importante, por reconhecer a natureza contextual do sentimento - um mesmo tweet pode expressar sentimentos diferentes dependendo da entidade em foco. Esta abordagem alinha-se com aplicações práticas de análise de sentimento, onde frequentemente busca-se compreender a percepção pública sobre entidades específicas.

III. Resultados

A comparação entre os modelos BiLSTM e BERT para classificação de sentimentos em tweets evidencia contrastes significativos entre uma arquitetura recorrente clássica e uma arquitetura baseada em atenção contextual moderna. O modelo BiLSTM, embora mais simples, demonstrou desempenho sólido, alcançando F1-Scores acima de 0.95 nas quatro classes avaliadas: Positive, Negative, Neutral e Irrelevant (Fig. 1). Isso mostra que redes recorrentes bidirecionais continuam sendo eficazes em tarefas de Processamento de Linguagem Natural (PLN), especialmente quando há um pré-processamento cuidadoso e uma arquitetura bem regularizada. Sua principal vantagem reside na leveza computacional: o BiLSTM treinou de forma rápida, utilizou menos memória e exigiu menos processamento, sendo ideal para aplicações em ambientes com recursos limitados.

Em contrapartida, o modelo BERT apresentou desempenho global ligeiramente superior, com acurácia de validação de 96,6% e F1-Scores consistentes em todas as classes, incluindo a categoria “Irrelevant”, considerada desafiadora por englobar mensagens ambíguas, neutras vagas ou sem polaridade clara (Fig. 1). O BERT se beneficia de seu pré-treinamento em grandes volumes de texto e da arquitetura baseada em atenção, o que permite capturar relações contextuais complexas com alta precisão. Sua principal força é a robustez semântica e a capacidade de gerar representações linguísticas mais profundas e contextualizadas. No entanto, essa sofisticação exige maior capacidade computacional, maior tempo de treinamento e o uso de aceleradores como GPUs ou TPUs, o que pode ser um obstáculo em determinados contextos de produção.

Os resultados obtidos confirmam claramente as características teóricas de cada abordagem. O BERT, mesmo com um fluxo de pré-processamento mais simples, alcançou ótimo desempenho (Fig. 3), reforçando sua capacidade de generalização e sua adaptabilidade a textos informais e ruidosos, como os presentes no Twitter. Por outro lado, o BiLSTM demandou um pipeline mais rigoroso — com limpeza textual agressiva, truncamento, padronização e codificação explícita de entidades — para atingir métricas próximas às do BERT (Fig. 2). Essa dependência maior do tratamento dos dados de entrada reflete a sensibilidade estrutural das redes recorrentes à qualidade linguística do corpus, especialmente em domínios informais.

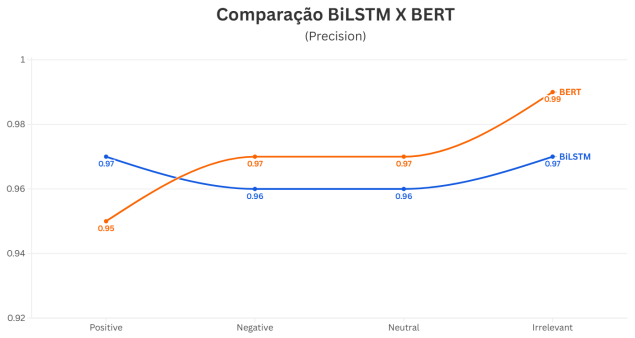

Fig. 1. Visualização gráfica comparativa dos resultados de precisão obtidos, pelos 2 modelos em estudo, para cada classe de análise.

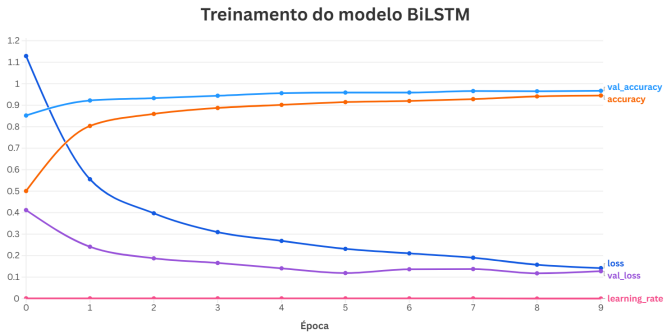

Fig. 2. Visualização gráfica dos dados obtidos durante o treinamento usando o modelo BiLSTM.

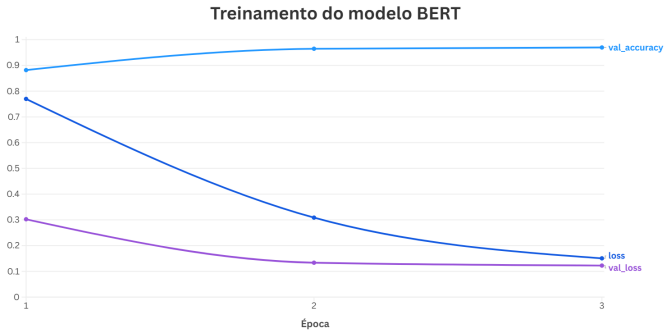

Fig. 3. Visualização gráfica dos dados obtidos durante o treinamento usando o modelo BERT.

IV. Conclusão

A escolha entre BiLSTM e BERT depende diretamente do contexto de aplicação. Para projetos que exigem máxima precisão com acesso a recursos computacionais robustos, o BERT é claramente a opção mais poderosa. Já em cenários onde simplicidade, agilidade e menor custo são prioritários, o BiLSTM se mostra uma alternativa viável e competitiva, com desempenho muito próximo ao do BERT. Ambos os modelos foram avaliados de forma completa, e demonstraram capacidade de lidar com os desafios impostos pela análise de sentimentos em redes sociais. Os dados experimentais corroboram essas conclusões ao mostrar o BERT com leve vantagem geral, mas sem comprometer a relevância prática do BiLSTM em aplicações mais restritas.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[2] B. Liu, "Sentiment Analysis and Opinion Mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, Morgan & Claypool Publishers, 2012.

[3] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," *arXiv preprint arXiv:1603.04467*, 2016.

[4] Y. Goldberg, "Neural Network Methods for Natural Language Processing," *Synthesis Lectures on Human Language Technologies*, vol. 10, no. 1, pp. 1–309, Morgan & Claypool Publishers, 2017.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017, pp. 5998–6008.

[6] J. Doe, "Twitter Entity Sentiment Analysis Dataset," Kaggle, [Online]. Available: https://www.kaggle.com/discussions/general/340511. Accessed on: Jun. 03, 2025.