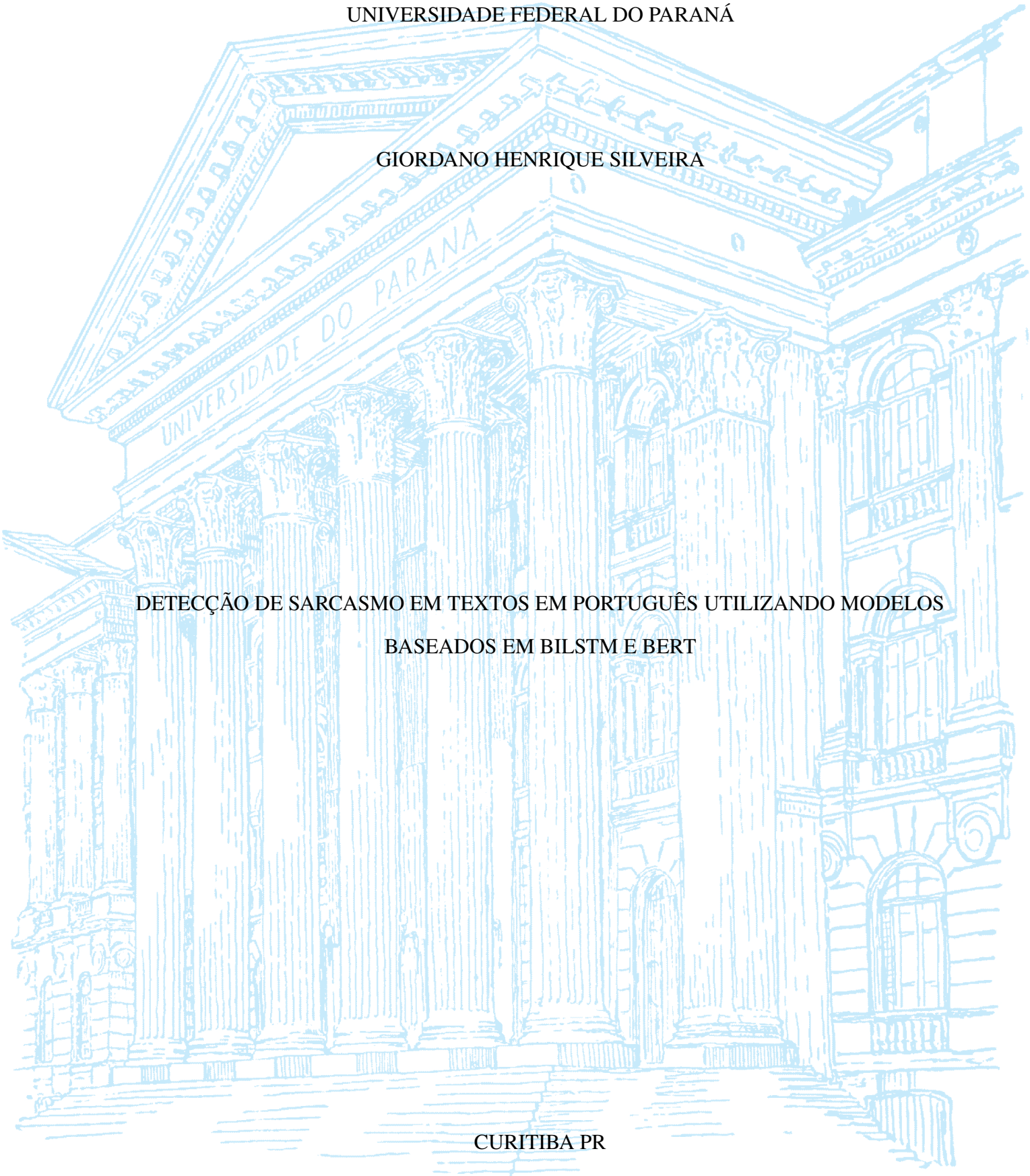
UNIVERSIDADE FEDERAL DO PARANÁ

GIORDANO HENRIQUE SILVEIRA



DETECÇÃO DE SARCASMO EM TEXTOS EM PORTUGUÊS UTILIZANDO MODELOS BASEADOS EM BILSTM E BERT

CURITIBA PR

2025

GIORDANO HENRIQUE SILVEIRA

DETECÇÃO DE SARCASMO EM TEXTOS EM PORTUGUÊS UTILIZANDO MODELOS BASEADOS EM BILSTM E BERT

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Computação*.

Orientador: Fabiano Silva.

CURITIBA PR

2025

RESUMO

O sarcasmo, presente no cotidiano, é frequentemente empregado para criticar, muitas vezes com humor, pessoas, instituições ou eventos. Por depender de contexto, conhecimento de mundo, tom e implicaturas pragmáticas, pode ser difícil de compreender até mesmo para seres humanos, já que o sentido pretendido geralmente contraria o literal. Consequentemente, a detecção automática de sarcasmo constitui um desafio relevante em Processamento de Linguagem Natural (PLN), pois erros na sua identificação podem distorcer tarefas a jusante, como análise de sentimento e mineração de opiniões. Nos últimos anos, o tema ganhou destaque justamente por seu impacto na qualidade dessas aplicações e pela necessidade de modelos capazes de captar nuances semânticas e contextuais. Neste trabalho, foram abordados dois modelos para a detecção de sarcasmo. O primeiro modelo é baseado em BiLSTM (Bidirectional Long Short-Term Memory), já o segundo modelo é baseado em BERT (Bidirectional Encoder Representations from Transformers), juntamente com BiLSTM. Como conjunto de dados, foram utilizadas manchetes sarcásticas retiradas do site The Onion e traduzidas para o português utilizando a API do ChatGPT. Ao final do trabalho, é feita uma comparação entre os dois modelos. Conclui-se que o modelo BERT com BiLSTM foi superior, alcançando F1 de 0,85.

Palavras-chave: Processamento de linguagem natural, Sarcasmo, BERT, LSTM.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Sarcasmo é uma forma de expressão frequentemente utilizada na comunicação humana, caracterizada por uma discrepância entre o significado literal das palavras e a intenção real do falante. No dicionário, sarcasmo é definido como "uma ironia insultuosa, penosa, mordaz ou cáustica"(dic, 2024). Ironia, por sua vez, é definida como "ação de dizer o oposto do que se quer expressar"(dic, 2024). A diferença entre sarcasmo e ironia é sutil, mas importante: enquanto a ironia pode ser usada de forma mais ampla para expressar uma incongruência entre o que é dito e o que é realmente pretendido, o sarcasmo geralmente envolve uma intenção mais direta de zombar ou criticar algo ou alguém. Sendo de difícil detecção até mesmo para humanos, o sarcasmo apresenta um desafio significativo para sistemas de processamento de linguagem natural (PLN). A detecção automática de sarcasmo é crucial para melhorar a compreensão de textos em diversas aplicações, por exemplo, em análises de sentimento.

Nos últimos anos, a detecção de sarcasmo tem sido abordada com o uso de técnicas de aprendizado de máquina, incluindo modelos baseados em redes neurais profundas como LSTM (Long Short-Term Memory) (Kumar et al., 2020), CNN (*Convolutional Neural Networks*) (Razali et al., 2021) e, mais recentemente, modelos pré-treinados como BERT (*Bidirectional Encoder Representations from Transformers*) (Shu, 2024). Estes modelos têm demonstrado um desempenho promissor na identificação de sarcasmo em textos, aproveitando grandes volumes de dados para capturar padrões linguísticos complexos.

Trabalhos anteriores na área têm explorado diversas abordagens para a detecção de sarcasmo e para diversas línguas, incluindo inglês (Scola e Segura-Bedmar, 2021), árabe (Rahma et al., 2023) e hindi (Swami et al., 2018). No entanto, a detecção de sarcasmo em português ainda é um campo relativamente inexplorado, com poucos estudos focados em textos brasileiros. De fato, não foram encontrados muitos trabalhos abordando especificamente a detecção de sarcasmo em português, o que destaca a necessidade de pesquisa adicional nesta área.

Este trabalho propõe uma abordagem para a detecção automática de sarcasmo em textos em português, utilizando modelos de aprendizado profundo, com ênfase em BERT e LSTM. A contribuição principal deste trabalho é o desenvolvimento e avaliação de modelos eficazes para a detecção de sarcasmo em português, além da criação de um conjunto de dados anotados especificamente para este propósito.

A base de dados utilizada neste trabalho é composta por manchetes de notícias sarcásticas, coletadas de fontes online: como o site *The Onion*, conhecido por seu conteúdo satírico. Esta base já tinha sido utilizada em trabalhos anteriores, como o de Nayak e Bolla (2022). A escolha desta base de dados se justifica primeiramente por estar disponível publicamente, além de conter textos gramáticalmente corretos e bem estruturados, o que facilita a análise e o desenvolvimento de modelos de detecção de sarcasmo.

Muitos trabalhos anteriores (Razali et al., 2021; Tan et al., 2023) focaram em textos curtos, como *tweets*, que frequentemente contêm erros gramaticais, abreviações, linguagem coloquials, *hashtags*, *emojis* e uma variedade de outros elementos que podem complicar a análise linguística. Além disso, as bases do twitter, ou de outras redes sociais, foram coletadas utilizando palavras-chave ou *hashtags* específicas. No twitter, por exemplo, usuários podem marcar suas postagens como sarcásticas usando *hashtags* como *#sarcasm* ou *#irony*. No entanto, nem todos os usuários utilizam essas hashtags de forma consistente, o que pode levar a uma representação incompleta do sarcasmo na base de dados

Em contraste, as manchetes de notícias sarcásticas são geralmente escritas por profissionais, resultando em textos mais claros e coerentes. Isso pode facilitar a detecção de sarcasmo, uma vez que os modelos podem se concentrar nas nuances linguísticas sem a interferência de ruídos comuns em textos de redes sociais. Portanto, a escolha de uma base de dados composta por manchetes de notícias sarcásticas permite explorar a detecção de sarcasmo em um contexto mais formal e estruturado, oferecendo *insights* valiosos para o desenvolvimento de modelos de PLN eficazes.

A base de dados, porém, apresenta limitações, como o fato de ser composta exclusivamente por textos em inglês. Portanto, mais uma das contribuições deste trabalho é a tradução e adaptação desta base de dados para o português, permitindo a avaliação dos modelos propostos em um contexto linguístico diferente. Como a tradução automática foi feita será explicado no Capítulo 4.

1.1 CONSIDERAÇÕES

A organização deste documento é a seguinte: o Capítulo 2 apresenta a fundamentação teórica sobre os principais conceitos e técnicas relacionados à detecção de sarcasmo e aos modelos de aprendizado profundo utilizados. O Capítulo 3 explora trabalhos relacionados na área de detecção de sarcasmo, destacando as abordagens e resultados obtidos. O Capítulo 4 detalha o modelo proposto, incluindo a arquitetura, os dados utilizados e o processo de treinamento. O Capítulo 5 apresenta os experimentos realizados, os resultados obtidos e a análise dos mesmos. Finalmente, o Capítulo 6 conclui o documento, resumindo as principais contribuições e sugerindo direções para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos teóricos fundamentais para o desenvolvimento deste trabalho. São abordados tópicos relacionados ao aprendizado de máquina, processamento de linguagem natural, redes neurais recorrentes e *Long Short-Term Memory* (LSTM), *Bidirectional Encoder Representations from Transformers* (BERT) e as métricas de avaliação empregadas. Esses conceitos são essenciais para compreender as técnicas e metodologias aplicadas na detecção de sarcasmo.

2.1 PROCESSAMENTO DE LINGUAGEM NATURAL

Processamento de Linguagem Natural (PLN) é um campo de pesquisa que tem como objetivo desenvolver sistemas e métodos para permitir que computadores compreendam, interpretem e gerem linguagem humana de forma automática (de Medeiros Caseli e das Graças Volpe Nunes, 2023). É "natural"no sentido de que a linguagem é humana, em oposição a linguagens de programação ou outras formas de comunicação estruturada.

Este campo de pesquisa divide-se em duas áreas principais (de Medeiros Caseli e das Graças Volpe Nunes, 2023): a interpretação da linguagem natural (ILN) e a geração de linguagem natural (GLN). A ILN concentra-se na compreensão, ou análise, e interpretação da linguagem humana, permitindo que os computadores extraíam significado e informações úteis a partir de textos ou fala. Uma aplicação comum da ILN é um *chatbot*, que pode entender perguntas feitas em linguagem natural e fornecer respostas relevantes. Já a GLN foca na criação de textos ou fala em linguagem humana de forma automática. Isso envolve a geração de textos coerentes e contextualmente apropriados, como resumos automáticos de documentos, geração de relatórios ou, usando novamente o exemplo do *chatbot*, a geração de respostas em linguagem natural para as perguntas dos usuários.

Neste trabalho, o foco está na ILN, especificamente na detecção de sarcasmo em títulos de notícias. Essa tarefa envolve a análise e interpretação de textos para identificar expressões sarcásticas, o que requer uma compreensão entre as palavras, o contexto e as nuances da linguagem utilizada. Para tal, modelos de aprendizado de máquina são treinados para capturar essas relações e realizar a classificação dos títulos como sarcásticos ou não sarcásticos. Modelos podem ser treinados do zero, ou podem ser utilizados modelos pré-treinados, que já possuem um conhecimento prévio sobre a linguagem, e são ajustados para a tarefa específica de detecção de sarcasmo. Treinar um modelo é atualizá-lo e uma das formas de fazer isso é o treinamento continuado.

O treinamento continuado (de Medeiros Caseli e das Graças Volpe Nunes, 2023) é treinar o modelo pré-treinado em uma nova base de dados, que é específica para a tarefa desejada que é diferente da base de dados utilizada no pré-treinamento, porém mantendo a mesma tarefa geral. A detecção de sarcasmo em títulos de notícias é uma tarefa específica dentro do campo mais amplo do processamento de linguagem natural: a análise de sentimentos. Portanto, o treinamento continuado é uma abordagem adequada para adaptar modelos pré-treinados para essa tarefa específica. Existem dois tipos de treinamento continuado: O treinamento continuado com foco na tarefa (TAPT - *Task Adaptive Pre-Training*) e o treinamento continuado com foco na adaptação ao domínio (DAPT - *Domain Adaptive Pre-Training*) (de Medeiros Caseli e das Graças Volpe Nunes, 2023).

O TAPT envolve a adaptação de um modelo pré-treinado para uma tarefa específica, utilizando uma base de dados não rotulada relacionada à tarefa. No caso da detecção de sarcasmo em títulos de notícias, o TAPT poderia ser realizado utilizando uma grande coleção de títulos de notícias, sem rótulos, para ajustar o modelo pré-treinado. Isso permitiria que o modelo aprendesse as características específicas dos títulos de notícias, melhorando sua capacidade de detectar sarcasmo nessa tarefa específica. Já o DAPT está preocupado com o domínio. Neste caso, o modelo é treinado por mais algum tempo em um domínio específico. Detecção de sarcasmo em títulos de notícias é uma tarefa que envolve o domínio das notícias, portanto, a abordagem utilizada neste trabalho foi o DAPT, onde o modelo pré-treinado é ajustado utilizando uma base de dados de títulos de notícias, tanto sarcásticos quanto não sarcásticos.

Treinar um modelo é uma tarefa de aprendizado de máquina, tópico este que será abordado na próxima seção. Nas seções seguintes, serão apresentados os modelos de aprendizado de máquina utilizados neste trabalho, Redes Neurais Recorrentes (RNNs) e *Long Short-Term Memory* (LSTM), e o modelo BERT.

2.2 APRENDIZADO DE MÁQUINA

De acordo com (Mitchell, 1997):

> "Um programa de computador aprende com a experiência E em relação a uma tarefa T e uma medida de desempenho P, se seu desempenho em T, medido por P, melhora com a experiência E."

Isto significa que um sistema de aprendizado de máquina é capaz de melhorar seu desempenho em uma tarefa específica à medida que é exposto a mais dados ou experiências relacionadas a essa tarefa.

O aprendizado de máquina pode ser dividido em três categorias principais (Alpaydin, 2014), aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço. No aprendizado supervisionado, o modelo é treinado com um conjunto de dados rotulado, onde cada exemplo de entrada está associado a uma saída desejada. O objetivo do modelo é aprender a mapear entradas para saídas corretas, de forma que possa fazer previsões precisas em novos dados. Já no aprendizado não supervisionado, o modelo é treinado com um conjunto de dados não rotulado, onde o objetivo é descobrir padrões ou estruturas subjacentes nos dados. Isso pode incluir tarefas como agrupamento (clustering) ou redução de dimensionalidade. Por fim, no aprendizado por reforço, o modelo aprende a tomar decisões sequenciais em um ambiente, recebendo recompensas ou punições com base em suas ações. O objetivo é aprender uma política que maximize a recompensa acumulada ao longo do tempo.

O Aprendizado Profundo (Goodfellow et al., 2016) é uma subárea do aprendizado de máquina que se concentra no uso de redes neurais profundas, ou seja, redes neurais com múltiplas camadas ocultas. Essas redes são capazes de aprender representações hierárquicas dos dados, permitindo que o modelo capture padrões complexos e relações não lineares. O Aprendizado Profundo tem sido particularmente bem-sucedido em tarefas como reconhecimento de imagem, processamento de linguagem natural e jogos, onde grandes quantidades de dados e poder computacional estão disponíveis.

Neste trabalho, o foco está no aprendizado supervisionado, onde modelos de Aprendizado Profundo são treinados para detectar sarcasmo em títulos de notícias. Os modelos utilizados incluem redes neurais recorrentes (LSTM) e o modelo BERT, ambos capazes de capturar o contexto e as nuances presentes na linguagem natural.

2.3 REDES NEURAIS RECORRENTES E LSTM

Redes Neurais Recorrentes (RNNs) são uma classe de redes neurais projetadas para lidar com dados sequenciais, como texto ou séries temporais. Ela funciona mantendo um estado interno que captura informações sobre elementos anteriores na sequência, permitindo que a rede aprenda dependências de longo prazo (Hochreiter e Kepler, 1997). No entanto, as RNNs tradicionais enfrentam desafios como o problema do desvanecimento do gradiente, que dificulta o aprendizado de dependências de longo prazo.

Para superar essas limitações, foram desenvolvidas variantes das RNNs, como as *Long Short-Term Memory* (LSTM) (Hochreiter e Kepler, 1997). As LSTMs introduzem uma arquitetura de célula especial que inclui portas de entrada, esquecimento e saída, permitindo que a rede controle o fluxo de informações e mantenha informações relevantes por períodos mais longos. Isso é particularmente útil para tarefas de processamento de linguagem natural, onde o contexto e a ordem das palavras são importantes para a compreensão do significado.

O modelo LSTM funciona da seguinte maneira (Hochreiter e Kepler, 1997): a porta de entrada decide quais informações da entrada atual devem ser armazenadas na célula de memória, a porta de esquecimento determina quais informações devem ser descartadas da célula de memória, e a porta de saída controla quais informações da célula de memória devem ser usadas para gerar a saída da rede. Essa arquitetura permite que as LSTMs aprendam dependências de longo prazo de forma mais eficaz do que as RNNs tradicionais. Isso porque as LSTMs são capazes de manter informações relevantes na célula de memória por períodos mais longos, enquanto as RNNs tradicionais tendem a esquecer informações importantes devido ao problema do desvanecimento do gradiente.

Visto que este trabalho visa detectar sarcasmo em títulos de notícias, é de suma importância capturar o contexto e as nuances presentes na linguagem e compreender como as palavras se relacionam entre si. Logo, para melhor capturar essas relações, foi utilizado o BiLSTM (*Bidirectional* LSTM). O BiLSTM é uma *Bidirectional Recurrent Neural Network* (BRNN) (Schuster e Paliwal, 1997), isto é, uma rede neural recorrente que processa a sequência de dados em ambas as direções, para frente e para trás. Isso permite que o modelo tenha acesso a informações contextuais tanto do passado quanto do futuro, melhorando sua capacidade de compreender o significado das palavras em um determinado contexto.

O exemplo a seguir ilustra o funcionamento do BiLSTM. Considere a frase "Ele não gostou do filme porque era muito longo". Ao processar essa frase, o LSTM unidirecional pode ter dificuldade em entender o significado da palavra "porque", pois ela depende do contexto fornecido pelas palavras que a seguem. No entanto, o BiLSTM pode capturar essa dependência, pois processa a frase em ambas as direções, permitindo que o modelo compreenda o significado completo da frase. Como este trabalho envolve a detecção de sarcasmo em títulos de notícias, o uso do BiLSTM é particularmente vantajoso, pois o sarcasmo muitas vezes depende do contexto e das nuances presentes na linguagem. Ao utilizar o BiLSTM, o modelo pode capturar melhor essas relações e melhorar sua capacidade de detectar sarcasmo.

O modelo BiLSTM foi utilizado em algumas vezes na literatura para a detecção de sarcasmo e apresentou resultados promissores. Isso será apresentado no próximo capítulo, onde serão discutidos os trabalhos relacionados a este tema.

2.4 BERT

O BERT (*Bidirectional Encoder Representations from Transformers*) (Devlin et al., 2018) é um modelo de linguagem baseado na arquitetura *Transformer* (Vaswani et al., 2017),

que foi pré-treinado em uma grande quantidade de dados textuais. Uma arquitetura *Transformer* é composta por camadas de atenção, que são camadas que permitem que o modelo foque em diferentes partes da entrada ao processar uma sequência, e camadas *feed-forward*, que são camadas totalmente conectadas que processam as representações intermediárias geradas pelas camadas de atenção. Essa arquitetura permite que o modelo capture o contexto bidirecional das palavras, o que o torna muito eficaz para tarefas de processamento de linguagem natural. O BERT consiste em camadas empilhadas de codificadores *Transformer* (*Transformer Encoders*), que são responsáveis por processar a entrada e gerar representações contextuais das palavras.

O BERT é pré-treinado utilizando duas tarefas principais (Devlin et al., 2018): *Masked Language Modeling* (MLM) e *Next Sentence Prediction* (NSP). No MLM, o modelo é treinado para prever palavras mascaradas em uma sequência de texto, isto é, algumas palavras são substituídas por um token especial [MASK], e o modelo deve prever quais palavras foram mascaradas com base no contexto fornecido pelas palavras restantes. No NSP, o modelo é treinado para prever se uma sentença B segue uma sentença A na sequência original do texto. Essas tarefas permitem que o BERT aprenda representações ricas das palavras e do contexto em que elas aparecem, o que o torna muito eficaz para umampla variedade de tarefas de processamento de linguagem natural.

O BERT utilizado neste trabalho é o BERTimbau (de Souza, 2020), que é uma versão do BERT treinada em português. O BERTimbau foi treinado usando dados de BrWaC , uma grande coleção de textos em português da web. Como resultado, o BERTimbau é capaz de capturar as nuances e particularidades da língua portuguesa, o que o torna especialmente adequado para tarefas de processamento de linguagem natural nessa língua.

A função base do BERT é gerar representações contextuais das palavras em uma sequência de texto. Essas representações são vetores densos que capturam o significado e o contexto das palavras, levando em consideração as palavras que as cercam. Essas representações podem ser utilizadas como entradas para outros modelos de aprendizado de máquina, como redes neurais, para realizar tarefas específicas de processamento de linguagem natural, função essa que foi utilizada neste trabalho e descrita nos capítulos seguintes.

2.5 MÉTRICAS DE AVALIAÇÃO

Métricas de avaliação são essenciais para medir o desempenho dos modelos de aprendizado de máquina. Elas fornecem uma maneira objetiva de comparar diferentes modelos e técnicas, permitindo identificar quais abordagens são mais eficazes para uma determinada tarefa. Neste trabalho, serão utilizadas as seguintes métricas para avaliar o desempenho dos modelos propostos: acurácia, precisão, *recall* e *F1-score*, sendo está última a métrica principal para comparação entre os modelos.

A acurácia é uma métrica que indica a proporção de previsões corretas em relação ao total de previsões feitas onde TP é o número de verdadeiros positivos, TN é o número de verdadeiros negativos, FP é o número de falsos positivos e FN é o número de falsos negativos. A acurácia é uma métrica útil quando as classes estão balanceadas, mas pode ser enganosa em casos de classes desbalanceadas.

$$\text{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.1}$$

A precisão é a proporção de verdadeiros positivos em relação ao total de previsões positivas feitas pelo modelo. A precisão é importante quando o custo de falsos positivos é alto.

$$\text{Precisão} = \frac{TP}{TP + FP} \tag{2.2}$$

O *recall*, também conhecido como sensibilidade, é a proporção de verdadeiros positivos em relação ao total de casos positivos reais, isto é, em relação ao total de exemplos que realmente pertencem à classe positiva, ele é crucial quando o custo de falsos negativos é alto.

$$Recall = \frac{TP}{TP + FN} \tag{2.3}$$

O *F1-score* é a média harmônica entre precisão e recall, fornecendo uma métrica balanceada que leva em conta ambos. *F1-score* é especialmente útil quando há um *trade-off*, ou seja, quando aumentar a precisão pode diminuir o recall e vice-versa. É desejável uma métrica que considere ambos os aspectos.

$$F1 = 2 \times \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}} \tag{2.4}$$

Esses métricas serão utilizadas para avaliar o desempenho dos modelos propostos na detecção de sarcasmo em títulos de notícias em português. Os resultados serão apresentados e discutidos no capítulo sobre avaliação experimental.

2.6 CONSIDERAÇÕES

Neste capítulo, foram apresentados os conceitos teóricos fundamentais para o desenvolvimento deste trabalho. Foram abordados tópicos relacionados ao processamento de linguagem natural, aprendizado de máquina, redes neurais recorrentes e LSTM, o modelo BERT, bem como as métricas de avaliação utilizadas. Esses conceitos são essenciais para compreender as técnicas e métodos aplicados na detecção de sarcasmo em títulos de notícias em português, que serão detalhadas nos capítulos seguintes.

3 TRABALHOS RELACIONADOS

Nesta seção, apresentamos uma revisão dos trabalhos relacionados ao tema abordado neste documento. Analisamos estudos anteriores que exploraram conceitos semelhantes, metodologias aplicadas e resultados obtidos, destacando as contribuições de cada um. A seguir, discutimos como esses trabalhos influenciaram o desenvolvimento do nosso estudo e como nosso trabalho se diferencia ou complementa as pesquisas existentes na área.

3.1 DEEP LEARNING PARA A DETECÇÃO DE SARCASMO EM TEXTOS

A detecção de sarcasmo em textos é um desafio significativo na área de processamento de linguagem natural (PLN). Vários estudos têm explorado o uso de técnicas de aprendizado profundo para abordar essa questão. Por exemplo, Kumar et al. (2020) combinaram redes neurais recorrentes (RNNs), mais especificamente uma BiLSTM, com mecanismos de atenção para focar em múltiplos aspectos semânticos de uma sentença. Para isto, ele utilizou duas bases de dados: Uma vinda do *Reddit*, uma rede social, (SARC) e outra, a qual ele denominou de IAC-V2, que contém debates informais extráidos da internet e foi rotulada por humanos. O modelo proposto por eles, obteve um desempenho um bom desempenho, chegando a alcançar um *F1-Score* de 77% sobre a base de dados do SARC, quando os dados estão balanceados.

Já Tan et al. (2023) propuseram um modelo baseado em *Deep Multi-Task Learning* para a detecção de sarcasmo e análise de sentimentos. Eles utilizam para isto uma arquitetura baseada em BiLSTM sobre uma base de dados do *Twitter*, além da mesma base utilizada por este trabalho, a de Notícias Sarcásticas. O modelo proposto por eles alcançou um F1-Score de 93% na detecção de sarcasmo, superando outros modelos de referência. Além disso, o modelo também demonstrou eficácia na análise de sentimentos, indicando a viabilidade do aprendizado multitarefa para essas tarefas relacionadas.

Continuando nessa linha, o trabalho apresentado por Razali et al. (2021) explorou o uso de redes neurais convolucionais (CNNs) para a detecção de sarcasmo em textos curtos, como *tweets*. Eles utilizaram uma base de dados composta por tweets rotulados como sarcásticos ou não sarcásticos. Seu modelo foi combinar uma CNN utilizando-a para extrair características do texto e, em seguida, alimentar um classificador de Regressão Logística com essas características. O modelo proposto alcançou um *F1-Score* de 95% na detecção de sarcasmo, superando outros métodos tradicionais de classificação.

Misra e Arora (2023a) foram quem criaram a base de dados de notícias sarcásticas utilizada neste trabalho. Eles propuseram um modelo baseado em CNN, BiLSTM e mecanismos de atenção para a detecção de sarcasmo em manchetes de notícias. O modelo proposto por eles alcançou uma acurácia de 90% na detecção de sarcasmo utilizando essa base de dados.

Por fim, Scola e Segura-Bedmar (2021) propuseram um modelo base que seria utilizado como baseline para a detecção de sarcasmo em textos. Eles utilizaram uma arquitetura baseada em BiLSTM com *embeddings* pré-treinados GloVe para representar as palavras. A base de dados utilizada por eles foi a mesma de notícias sarcásticas utilizada neste trabalho. O modelo alcançou um F1-Score de 86%.

3.2 BERT E SUAS VARIAÇÕES PARA TAREFAS DE CLASSIFICAÇÃO DE TEXTO

O BERT (*Bidirectional Encoder Representations from Transformers*) revolucionou o campo do processamento de linguagem natural ao introduzir uma abordagem baseada em transformers para o pré-treinamento de modelos de linguagem. Desde sua introdução, várias variações do BERT foram desenvolvidas para melhorar o desempenho em tarefas específicas de classificação de texto.

Baruah et al. (2020) exploraram o uso do BERT como *embedding* para a detecção de sentimentos em textos. Utilizando um BiLSTM sobre os embeddings gerados pelo BERT. Seria um trabalho parecido com os que serão apresentados a seguir, o que o diferencia é que eles utilizaram uma base de dados vinda do *Twitter*, composta não só pela resposta sarcástica, mas também contendo toda a conversa que a originou. O resultado é supreendente: quando o modelo considerou a resposta imediatamente anterior à resposta sarcástica, o modelo alcançou um *F1-Score* de 74%, e a medida que mais contexto era adicionado, pior era o desempenho. Quando adicionado 2 ou 3 respostas anteriores, o desempenho caiu para 50% e 33%, respectivamente. Se considerado todo o contexto da conversa, o desempenho melhora, porém não chega a ser tão bom quanto considerar apenas a resposta anterior, alcançando um *F1-Score* de 73%.

Nayak e Bolla (2022) exploraram o uso de diferentes técnicas de embeddings de palavras para melhorar o desempenho do BERT em tarefas de classificação de texto. Utilizando *TF-IDF*, *Word2Vec*, *Doc2Vec* e BERT como técnicas de *embeddings* e combinando-os com outros sete classificadores, sendo eles Naive Bayes, Regressão Logística, MLP, SVM, ELM, LSTM e BiLSTM, eles avaliaram o desempenho na mesma base de dados de notícias sarcásticas utilizada neste trabalho. O melhor desempenho foi obtido utilizando o BERT combinado com LSTM/BiLSTM, alcançando um *F1-Score* de 89%.

Em outro estudo, Shu (2024) comparou o desempenho entre BERT e sua variação RoBERTa (*Robustly Optimized BERT Pretraining Approach*) em tarefas de classificação de texto. Utilizando uma base de dados vinda do Reddit, eles avaliaram o desempenho dos dois modelos utilizando as métricas típicas de classificação. Eles utilizaram o BERT base, O BERT large, o RoBERTa base e o RoBERTa large. O melhor desempenho foi obtido utilizando o RoBERTa large, alcançando um *F1-Score* de 76

Já no trabalho de Khan et al. (2025) mais recentemente, foi proposto um modelo híbrido que combina RoBERTa com BiLSTM e mecanismos de atenção para melhorar a detecção de sarcasmo em textos. Eles utilizaram três bases de dados: Duas vindas do Reddit e uma do Twitter. O modelo proposto alcançou um *F1-Score* de 93% na detecção de sarcasmo, superando outros modelos de referência.

Pawestri et al. (2024) propuseram dois modelos baseados em RoBERTa para a detecção de sarcasmo em textos. Os dois modelos utilizaram RoBERTa como base, porém um modelo utilizou CNN e o outro uma BRNN. Utilizando uma base de notícias sarcásticas, o modelo baseado em CNN alcançou um *F1-Score* de 88%, enquanto o modelo baseado em BRNN alcançou um *F1-Score* de 65%.

Por fim, os mesmos Scola e Segura-Bedmar (2021) também exploraram o uso do BERT para a detecção de sarcasmo em textos. Eles utilizaram o BERT Large como classificador direto, sem a utilização de camadas adicionais. A base de dados utilizada por eles foi a mesma de notícias sarcásticas utilizada neste trabalho e o modelo alcançou um F1-Score de 90%.

3.3 CONSIDERAÇÕES

O modelo base proposto por (Scola e Segura-Bedmar, 2021), o BiLSTM, será utilizado neste trabalho também como baseline para comparação com os modelo proposto. O trabalho dele foi escolhido pois é um dos poucos que forneceu acesso ao código fonte e à base de dados utilizada, o que facilita a reprodução dos resultados. Além disso, o modelo proposto por ele é simples e eficiente, o que o torna uma boa referência para comparação com outros modelos mais complexos.

O modelo base de BiLSTM utilizado neste trabalho (Scola e Segura-Bedmar, 2021) é composto por uma camada de *embeddings*, que converte as palavras em vetores densos de tamanho fixo. Para isto, cada manchete foi tokenizada e convertida em uma sequência de índices inteiros. Cada token é mapeado para um vetor de *embeddings*, utilizando o FastText (Joulin et al., 2016) como *embeddings* pré-treinados. Os *embeddings* são necessários para que o modelo possa trabalhar com representações numéricas das palavras, permitindo que ele aprenda padrões e relações entre as palavras.

O FastText é um modelo de aprendizado de máquina desenvolvido pelo Facebook AI Research (FAIR) para gerar representações vetoriais de palavras (*word embeddings*) de maneira eficiente. Diferente de métodos tradicionais como *Word2Vec*, o *FastText* representa cada palavra como um conjunto de subpalavras (n-grams de caracteres). Isso permite que o modelo capture informações morfológicas e produza vetores mais robustos, especialmente para línguas ricas em flexão ou palavras raras. Além disso, o *FastText* disponibiliza (Grave et al., 2018) vetores pré-treinados em grandes corpora, como *Wikipedia* e *Common Crawl*, cobrindo mais de 150 idiomas. Esses *embeddings* podem ser reutilizados diretamente em tarefas de PLN, reduzindo o custo de treinamento e melhorando a qualidade do modelo em cenários com poucos dados.

Feitos os embeddings, as sequências de vetores são alimentadas em uma camada BiLSTM, que processa a sequência em ambas as direções (para frente e para trás). As saídas das duas direções são concatenadas, fornecendo uma representação rica do contexto de cada palavra na sequência. Em seguida, as saídas processadas passam por uma camada de *Global Max Pooling* que extrai as características mais relevantes e fortes ao longo de todas as etapas da sequência, condensando-as em um vetor de tamanho fixo. A representação resultante passa por camadas Dense, que interpretam, refinam e comprimem os recursos aprendidos. Para melhorar a generalização e reduzir o sobreajuste, o famoso *overfitting*, camadas de *Dropout* desativam neurônios aleatoriamente durante o treinamento. Por fim, uma camada de saída *Sigmoid* produz uma probabilidade entre 0 e 1, permitindo que o modelo realize uma classificação binária, no caso, sarcasmo ou não sarcasmo.

No próximo capítulo, apresentaremos o modelo proposto neste trabalho, que combina o BERT com a arquitetura BiLSTM, detalhando sua implementação e as motivações por trás dessa escolha.

4 MODELO PROPOSTO

Neste capítulo, será apresentado o modelo proposto para a detecção de sarcasmo em títulos de notícias em português. O modelo combina a arquitetura BiLSTM com o BERT como extrator de características, visando aproveitar o poder do BERT para capturar o contexto e as nuances da linguagem, enquanto o BiLSTM é responsável por aprender as relações temporais e sequenciais dos dados.

4.1 FERRAMENTAS E BIBLIOTECAS

Os experimentos realizados neste trabalho foram feitos no Google Colab, que é uma plataforma de computação em nuvem que permite a execução de código Python em *notebooks Jupyter*. O Google Colab oferece acesso a GPUs, o que é essencial para o treinamento de modelos de aprendizado profundo, como o LSTM e o BERT. Além disso, o Google Colab oferece uma série de bibliotecas pré-instaladas, como TensorFlow, Keras e *Transformers*, que facilitam o desenvolvimento e a implementação dos modelos. Como ambiente de desenvolvimento, foi utilizado o Jupyter *Notebook*, que é uma ferramenta interativa que permite a criação e compartilhamento de documentos que contêm código, visualizações e texto explicativo. O Jupyter Notebook é amplamente utilizado na comunidade de ciência de dados e aprendizado de máquina devido à sua facilidade de uso e flexibilidade. Para a implementação dos modelos LSTM e BERT, foram utilizadas as bibliotecas TensorFlow e Keras.

O TensorFlow é uma biblioteca de código aberto desenvolvida pelo Google para computação numérica e aprendizado de máquina. Ele oferece uma ampla gama de ferramentas e recursos para a construção, treinamento e implantação de modelos de aprendizado de máquina, incluindo suporte para redes neurais profundas e computação distribuída. O TensorFlow é amplamente utilizado na comunidade de aprendizado de máquina devido à sua flexibilidade, escalabilidade e desempenho. Já o Keras é uma biblioteca de alto nível construída sobre o TensorFlow, que fornece uma interface simples e intuitiva para a construção e treinamento de modelos de aprendizado profundo. O Keras oferece uma série de camadas pré-construídas, otimizadores e funções de perda, o que facilita o desenvolvimento de modelos complexos.

Além disso, para a utilização do modelo BERT, foi utilizada a biblioteca Transformers, desenvolvida pela Hugging Face. A biblioteca Transformers oferece uma ampla gama de modelos pré-treinados baseados na arquitetura Transformer, incluindo o BERT, GPT e RoBERTa. Ela fornece uma interface simples para o carregamento e utilização desses modelos, bem como ferramentas para o *fine-tuning* e avaliação. A biblioteca Transformers é amplamente utilizada na comunidade de processamento de linguagem natural devido à sua facilidade de uso e à qualidade dos modelos pré-treinados disponíveis. Como otimizador, foi utilizado o Adam (*Adaptive Moment Estimation*), que é um dos otimizadores mais populares e eficazes para o treinamento de redes neurais profundas.

Além disso, foi também utilizado o Pandas e o NumPy. O Pandas é uma biblioteca de código aberto que fornece estruturas de dados e ferramentas de análise de dados para a linguagem Python. Ele é amplamente utilizado na comunidade de ciência de dados devido à sua facilidade de uso e eficiência no manuseio de grandes conjuntos de dados. O NumPy é uma biblioteca fundamental para computação científica em Python, fornecendo suporte para arrays multidimensionais e uma ampla gama de funções matemáticas e estatísticas. Ele é amplamente

utilizado na comunidade de ciência de dados e machine learning devido à sua eficiência e desempenho em operações numéricas.

A seguir as versões das principais bibliotecas utilizadas neste trabalho:

- TensorFlow: 2.19.0
- Pandas: 2.2.2
- NumPy: 2.0.2
- Scikit-learn: 1.6.1
- Transformers: 4.57.3
- MatPlotLib: 3.10.0

4.2 BASE DE DADOS

A base de dados utilizada neste trabalho foi criada por (Misra e Arora, 2023b) e é composta por títulos de notícias extraídos do site TheOnion.com [1], conhecido por seu conteúdo satírico e sarcástico, e do site Huffington Post [2], que apresenta notícias sérias e informativas. Esta base de dados está disponível publicamente e pode ser acessada através do repositório no GitHub [3]. O propósito do site *The Onion* é entreter os leitores com notícias fictícias que utilizam o sarcasmo como principal recurso humorístico, isto prove um ambiente ideal para coletar exemplos de sarcasmo em títulos de notícias.

Estudos anteriores 3, sobre detecção de sarcasmo utilizavam bases de dados compostas por *tweets*, que são mensagens curtas postadas na plataforma *Twitter*. No entanto, *tweets* podem conter abreviações, gírias, erros gramaticais e outros elementos que dificultam a análise linguística. Além disso, muitas dassas bases foram coletadas de forma automática, utilizando *hashtags* como *#sarcasm* para identificar mensagens sarcásticas, porém isso de nada garante que todas as mensagens com essa hashtag sejam realmente sarcásticas, ou que todas as mensagens sarcásticas estejam marcadas com essa hashtag. Por outro lado, títulos de notícias são geralmente mais formais e estruturados, o que facilita a análise linguística e a detecção de sarcasmo. Portanto, a utilização de títulos de notícias como base de dados para detecção de sarcasmo pode proporcionar resultados mais precisos e confiáveis.

A base de dados contém 28.503 títulos de notícias, sendo 14.951 títulos sarcásticos do The Onion e 13.552 títulos não sarcásticos do *Huffington Post*, ou seja, a base de dados está balanceada entre as duas classes. Cada título é acompanhado por um rótulo binário, onde 1 indica que o título é sarcástico e 0 indica que o título não é sarcástico. A seguir, são apresentados alguns exemplos de títulos sarcasticos e não sarcásticos presentes na base de dados:

- Título sarcástico: "*ford develops new suv that runs purely on gasoline*";
- Título não sarcástico: "*what to know regarding current treatments for ebola*";
- Título sarcástico: "*mother comes pretty close to using word 'streaming' correctly*";
- Título não sarcástico: "*5 ways to file your taxes with less stress*".

[1] `https://www.theonion.com/`

[2] `https://www.huffpost.com/`

[3] `https://raw.githubusercontent.com/rishabhmisra/News-Headlines-Dataset-For-Sarcasm-Detection/refs/heads/master/Sarcasm_Headlines_Dataset.json`

- Título sarcástico: "*San Diego zoo displays first rhino stillborn in captivity*";
- Título não sarcástico: "*This D.C. restaurant just sued Trump and his hotel for unfair competition*";
- Título sarcástico: "*horrified geologists uncover millions of rocks in sprawling mass grave*";
- Título não sarcástico: "*top aide denies that donald trump posed as his own spokesman*".
- Título sarcástico: "*brilliant, innovative ceo just wrote words 'social media' on whiteboard and underlined it*";
- Título não sarcástico: "*bartender accused of plotting to poison John Boehner*";
- Título sarcástico: "*paramedics didn't realize how hard it would be to cut drunk woman out of elmo costume*";
- Título não sarcástico: "*first latino arab-american running for congress views his heritage as an asset*".

Esses exemplos ilustram a diferença entre títulos sarcásticos, que utilizam humor e ironia para transmitir uma mensagem, e títulos não sarcásticos, que são informativos e diretos.

Este trabalho é focado na detecção de sarcasmo em títulos de notícias em português. Portanto, a base de dados original, que está em inglês, foi traduzida para o português utilizando a API do ChatGPT [4]. O modelo escolhido para a tradução foi o GPT-4.1 Mini, que é capaz de realizar traduções de boa qualidade, entretanto, é importante ressaltar que a tradução automática pode introduzir erros ou nuances que não estavam presentes no texto original. Todos os cuidados foram tomados para garantir que a tradução mantivesse o significado e o contexto dos títulos originais. O *prompt* utilizado para a tradução foi elaborado para solicitar uma tradução literal, sem tentar interpretar o sarcasmo, de forma a preservar o sentido original dos títulos. O *prompt* utilizado foi o seguinte:

> "You are a professional translator. Translate the following English headlines into Brazilian Portuguese. Rules: 1. Preserve the original meaning, tone, and style. 2. If the sentence contains sarcasm, irony, exaggeration, or humor, keep it intact in Portuguese. 3. Do not explain or interpret sarcasm — just translate it naturally. 4. Keep proper names, places, and organizations unchanged. 5. Return only the translations, numbered correspondingly."

Além disso, no momento da criação, o modelo foi configurado com uma temperatura de 0.3, o que torna o modelo mais conservador, reduzindo a criatividade na resposta.

[4] `https://openai.com/blog/chatgpt`

```
response = client.chat.completions.create(
    model=model,
    messages=[{"role": "system",
               "content": "You are a professional translator."},
              {"role": "user", "content": prompt}],
    temperature=0.3
)
```

A coluna de títulos traduzidos, chamada "headline_pt", foi então adicionada à base de dados original, ela foi utilizada como entrada para os modelos de detecção de sarcasmo. A seguir, são apresentados alguns exemplos de títulos traduzidos para o português:

- Título sarcástico traduzido: "*Ford desenvolve novo SUV que funciona apenas com gasolina*";
- Título não sarcástico traduzido: "*O que saber sobre os tratamentos atuais para Ebola*";
- Título sarcástico traduzido: "*Mãe chega bem perto de usar a palavra 'streaming' corretamente*";
- Título não sarcástico traduzido: "*5 maneiras de declarar seus impostos com menos estresse*".
- Título sarcástico traduzido: "*Zoológico de san diego adquire homem chinês*";
- Título não sarcástico traduzido: "*Restaurante de D.C. processa Trump e seu hotel por concorrência desleal*";
- Título sarcástico traduzido: "*Geólogos horrorizados descobrem milhões de pedras em uma enorme cova comum*";
- Título não sarcástico traduzido: "*Assessor nega que Donald Trump se passou por seu próprio porta-voz*".
- Título sarcástico traduzido: "*Ceo brilhante e inovador acaba de escrever 'mídia social' no quadro e sublinhou*";
- Título não sarcástico traduzido: "*Bartender acusado de planejar envenenar John Boehner*";
- Título sarcástico traduzido: "*Paramédicos não perceberam como seria difícil tirar mulher bêbada de fantasia do elmo*";
- Título não sarcástico traduzido: "*Primeiro latino árabe-americano concorrendo ao congresso vê sua herança como um ativo*".

Analisando os exemplos de títulos traduzidos, é possível observar que o sarcasmo foi preservado na maioria dos casos, mantendo o tom humorístico e irônico dos títulos originais. A headline "*Ford desenvolve novo SUV que funciona apenas com gasolina*", por exemplo, mantém o sarcasmo ao sugerir que um SUV que funciona apenas com gasolina é algo inovador, o que é irônico dado o contexto atual de busca por veículos mais sustentáveis. Da mesma forma, o título "*Mãe chega bem perto de usar a palavra 'streaming' corretamente*"preserva o tom humorístico ao destacar a dificuldade de uma mãe em utilizar corretamente um termo tecnológico moderno.

Esses exemplos ilustram como a tradução automática, quando bem orientada, pode preservar nuances como o sarcasmo, que são essenciais para a compreensão do texto.

Se observamos os títulos não sarcásticos, como "*O que saber sobre os tratamentos atuais para Ebola*"e "*5 maneiras de declarar seus impostos com menos estresse*", percebemos que a tradução também manteve o tom informativo e direto, sem introduzir elementos de humor ou ironia. Isso é crucial para garantir que a distinção entre títulos sarcásticos e não sarcásticos seja clara, permitindo que os modelos de detecção de sarcasmo aprendam a identificar essas diferenças de forma eficaz.

Agora uma análise da base de dados traduzida revelou que os títulos possuem em média 10 a 12 palavras, com alguns títulos mais longos chegando a ter até 25 palavras. Em termos de caracteres, os títulos tem em média entre 50 a 70 caracteres, com alguns títulos mais longos chegando a ter até 150 caracteres. A seguir, são apresentados dois gráficos que ilustram a distribuição do número de palavras e caracteres por título na base de dados traduzida.

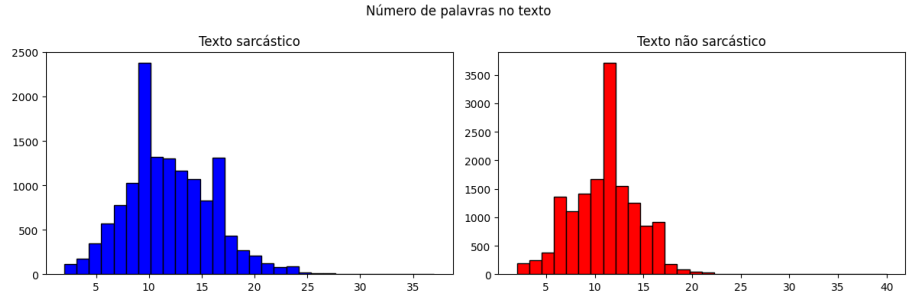


Figura 4.1: Distribuição do número de palavras por título na base de dados traduzida.

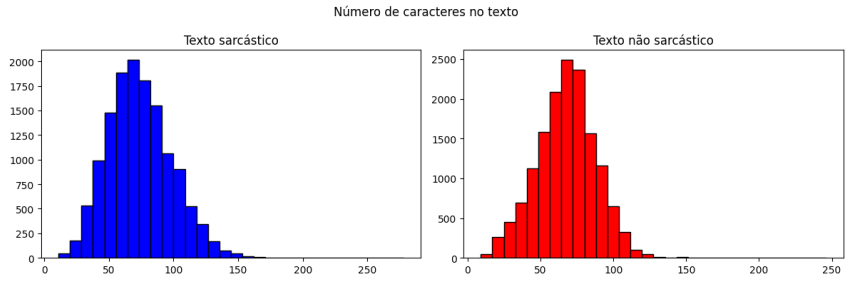


Figura 4.2: Distribuição do número de caracteres por título na base de dados traduzida.

Saber o número médio de palavras e caracteres por título é importante para o pré-processamento dos dados e para a configuração dos modelos de aprendizado profundo, como o BiLSTM e o BERT-BiLSTM, que serão utilizados neste trabalho. Isso ajuda a determinar o tamanho máximo das sequências de entrada e a escolher as técnicas adequadas.

A base de dados utilizada neste trabalho é composta por títulos de notícias sarcásticas e não sarcásticas, A base de dados então dividida em conjuntos de treino, validação e teste, com proporções de 70%, 15% e 15%, respectivamente. Essa divisão permite avaliar o desempenho do modelo de forma adequada, garantindo que o modelo seja treinado em uma parte dos dados e avaliado em dados não vistos durante o treinamento. Para a divisão dos dados, foi utilizado a função `train_test_split` do Scikit-learn. A divisão foi feita de forma estratificada, garantindo que a proporção de títulos sarcásticos e não sarcásticos seja mantida em cada conjunto, além de definir um valor fixo para o parâmetro `random_state`, garantindo a reprodutibilidade dos resultados.

4.3 MODELO BASE

Após a preparação dos dados, foi implementado o modelo BiLSTM. Primeiramente, os dados foram divididos em conjuntos de treino, validação e teste, conforme descrito anteriormente. Para avaliar o desempenho do modelo, a base de teste foi utilizada e os resultados são apresentados no Capítulo 5. O modelo BiLSTM foi implementado utilizando a API Sequencial do Keras. A arquitetura do modelo é composta por uma camada de *embedding*, seguida por uma camada BiLSTM com 128 unidades, seguida de uma operação de *Global Max Pooling*, seguidas por 2 camadas densas com funções de ativação ReLU, intercaladas com camadas de *dropout* para evitar sobreajuste. Finalmente, a camada de saída é uma camada densa com uma unidade e função de ativação sigmóide, que produz uma saída entre 0 e 1, indicando a probabilidade da manchete ser sarcástica.

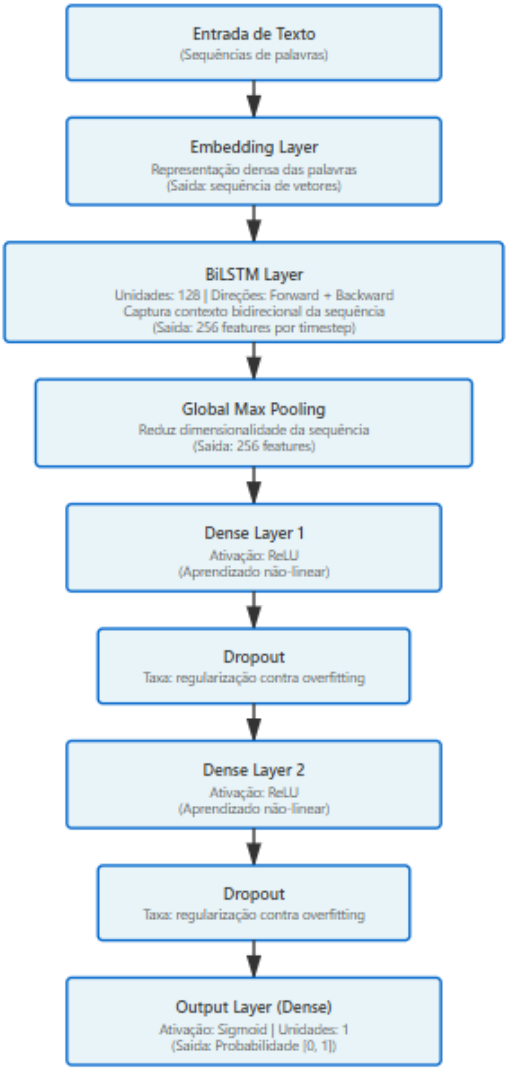


Figura 4.3: Arquitetura do modelo BiLSTM utilizado como baseline.

A operação de *Global Max Pooling* é utilizada para reduzir a dimensionalidade dos dados, condensando as informações mais relevantes em um vetor de tamanho fixo. Essa operação seleciona o valor máximo ao longo de cada dimensão, capturando as características mais fortes

presentes na sequência de saída da camada BiLSTM. As camadas densas subsequentes interpretam e refinam essas características, preparando-as para a classificação final. A função de ativação ReLU (*Rectified Linear Unit*) é amplamente utilizada em redes neurais devido à sua capacidade de introduzir não linearidades no modelo, permitindo que ele aprenda padrões complexos nos dados. As camadas de *dropout* são uma técnica de regularização que ajuda a prevenir o sobreajuste, desativando aleatoriamente uma fração dos neurônios durante o treinamento, o que força o modelo a aprender representações mais robustas. Por fim, a camada de saída com função de ativação sigmóide é adequada para problemas de classificação binária, produzindo uma probabilidade que pode ser interpretada como a confiança do modelo na classificação do título como sarcástico ou não sarcástico. O diagrama da arquitetura do modelo BiLSTM é apresentado na Figura 4.3.

A função de perda utilizada foi a *binary crossentropy*, adequada para problemas de classificação binária. Essa função mede a diferença entre as distribuições de probabilidade previstas pelo modelo e as distribuições reais dos rótulos, penalizando previsões incorretas de forma mais severa. A métrica utilizada para avaliar o desempenho do modelo durante o treinamento foi a acurácia (accuracy), que indica a proporção de previsões corretas feitas pelo modelo em relação ao total de previsões. O otimizador utilizado foi o Adam, com uma taxa de aprendizado de 0.00001. O Adam é um dos otimizadores, algoritmo que ajusta os pesos do modelo durante o treinamento, com o objetivo de minimizar a função de perda. É um dos mais populares e eficazes para o treinamento de redes neurais profundas.

O modelo foi treinado por um máximo de 25 épocas, com tamanho de lote (*batch size*) de 100. Durante o treinamento, foi utilizado o *Early Stopping*, que monitora a métrica de validação (val_loss) e interrompe o treinamento se a métrica não melhorar por 3 épocas consecutivas, prevenindo sobreajuste.

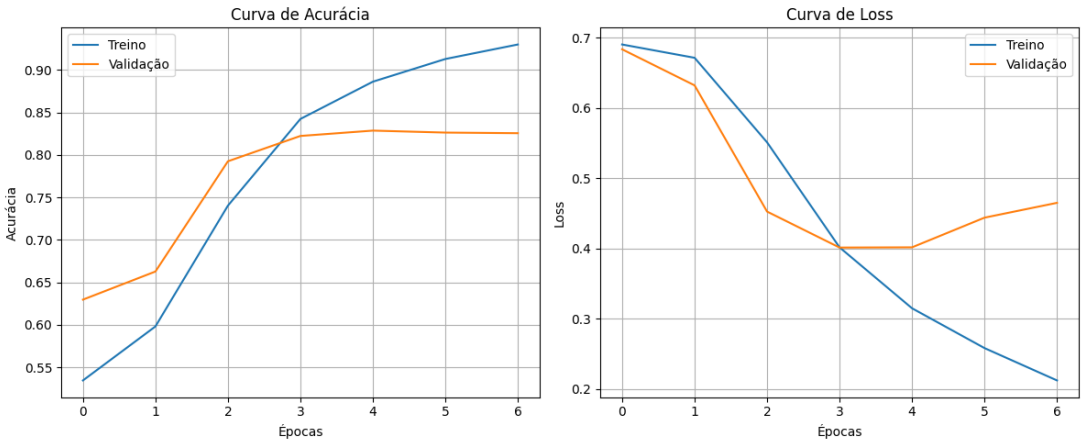


Figura 4.4: Gráfico de acurácia e perda durante o treinamento do modelo BiLSTM.

A figura 4.4 apresenta os gráficos de acurácia e perda (loss) durante o treinamento do modelo BiLSTM. A linha azul representa a métrica no conjunto de treinamento, enquanto a linha laranja representa a métrica no conjunto de validação. Observa-se que a acurácia do conjunto de treinamento aumenta ao longo de 3 épocas, atingindo um valor próximo de 81%, enquanto a acurácia do conjunto de validação atinge um valor máximo de aproximadamente 82% na terceira época, indicando que o modelo está aprendendo a classificar corretamente os títulos sarcásticos e não sarcásticos. Após a terceira época a acurácia do conjunto de validação começa a diminuir, indicando que o modelo está começando a sobreajustar os dados de treinamento. A perda (*loss*) do conjunto de treinamento diminiui ao longo das épocas, indicando que o modelo está melhorando sua capacidade de prever os rótulos corretos, enquanto a perda do conjunto de

validação atinge um valor mínimo na terceira época, após o qual começa a aumentar, reforçando a indicação de sobreajuste. O *Early Stopping* foi acionado após a sexta época, visto que a métrica de validação não melhorou por 3 épocas consecutivas, previnindo o sobreajuste e garantindo que o modelo mantenha uma boa capacidade de generalização.

4.4 MODELO PROPOSTO: BERT-BILSTM

Além do modelo BiLSTM, foi implementado o modelo BERT-BiLSTM, que utiliza o modelo pré-treinado BERT como camada de *embedding*. Como descrito anteriormente, o BERT é um modelo de linguagem baseado na arquitetura *Transformer*, que foi pré-treinado em uma grande quantidade de dados textuais. O BERT é capaz de capturar o contexto bidirecional das palavras, o que o torna muito eficaz para tarefas de processamento de linguagem natural.

Primeiramente, os títulos foram tokenizados utilizando o tokenizador do BERTimbau, que converte as palavras em tokens compatíveis com o modelo. Esses tokens foram então alimentados no modelo, que gerou representações vetoriais para cada título. Essas representações foram então utilizadas como entradas para o modelo BiLSTM, que realizou a classificação dos títulos em sarcásticos ou não sarcásticos. A arquitetura do modelo BERT-BiLSTM é semelhante à do modelo BiLSTM, com a diferença de que a camada de *embedding* é substituída pelo modelo BERT pré-treinado. A função de perda, otimizador, tamanho de lote e estratégia de *Early Stopping* utilizados foram os mesmos do modelo BiLSTM.

Espera-se que a incorporação do BERT como camada de embedding traga melhorias no desempenho do modelo, especialmente na captura do contexto e das nuances da linguagem presentes nos títulos de notícias.

4.5 CONSIDERAÇÕES

Neste capítulo, foram apresentados o modelo proposto para a detecção de sarcasmo em títulos de notícias em português, bem como as ferramentas e bibliotecas utilizadas para a implementação dos modelos. Foi descrita a base de dados utilizada, incluindo o processo de tradução dos títulos para o português. Além disso, foram detalhados o modelo base BiLSTM e o modelo proposto BERT-BiLSTM, incluindo suas arquiteturas e configurações de treinamento.

O modelo BiLSTM serve como uma linha de base para comparação com o modelo proposto, permitindo avaliar o impacto da incorporação do BERT como camada de *embedding*. Está arquitetura visa aproveitar o poder do BERT para capturar o contexto e as nuances da linguagem, enquanto o BiLSTM é responsável por aprender as relações temporais e sequenciais dos dados. Está mesma arquitetura foi utilizada em trabalhos anteriores - capítulo 3 - e apresentou bons resultados na detecção de sarcasmo em textos. Portanto, espera-se que o modelo proposto apresente um desempenho superior ao modelo base, especialmente na captura do contexto e das nuances da linguagem presentes nos títulos de notícias.

No próximo capítulo, serão apresentados os experimentos realizados para avaliar o desempenho do modelo proposto, bem como uma comparação com outros modelos existentes na literatura.

5 AVALIAÇÃO EXPERIMENTAL

Neste capítulo, serão apresentados os experimentos realizados para avaliar o desempenho do modelo proposto na detecção de sarcasmo em títulos de notícias em português. Serão descritos os dados utilizados, os resultados obtidos e uma análise comparativa com outros modelos existentes na literatura.

5.1 MÉTRICAS DE AVALIAÇÃO

Como descrito no Capítulo 2, para avaliar o desempenho do modelo proposto, foram utilizadas as seguintes métricas:

- Acurácia: Mede a proporção de previsões corretas em relação ao total de previsões realizadas.
- Precisão: Mede a proporção de verdadeiros positivos em relação ao total de positivos previstos pelo modelo.
- *Recall*: Mede a proporção de verdadeiros positivos em relação ao total de positivos reais no conjunto de dados.
- *F1-Score*: É a média harmônica entre precisão e *recall*, fornecendo uma medida balanceada do desempenho do modelo.

Essas métricas foram escolhidas devido à sua capacidade de fornecer uma visão abrangente do desempenho do modelo, considerando tanto a capacidade de identificar corretamente os exemplos positivos quanto a capacidade de evitar falsos positivos. A principal métrica utilizada para comparar o desempenho dos modelos foi o *F1-Score*, pois ele leva em conta tanto a precisão quanto o recall. Um *F1-Score* mais alto indica tanto uma alta precisão quanto um alto *recall*, refletindo um desempenho geral melhor do modelo na tarefa de detecção de sarcasmo.

5.2 RESULTADOS OBTIDOS

Nesta seção, serão apresentados os resultados obtidos pelos experimentos realizados com o modelo proposto, bem como uma comparação com outros modelos existentes na literatura. Os resultados serão apresentados em termos das métricas de avaliação descritas anteriormente. A Tabela 5.1 apresenta um resumo dos resultados obtidos pelos diferentes modelos na tarefa de detecção de sarcasmo. Os trabalhos listados na tabela são diferentes artigos da literatura que abordaram a detecção de sarcasmo utilizando diversas abordagens e bases de dados. Eles foram incluídos para fornecer um contexto comparativo para os resultados do modelo proposto neste trabalho. O símbolo •indica que o valor exato da métrica não foi reportado no trabalho original.

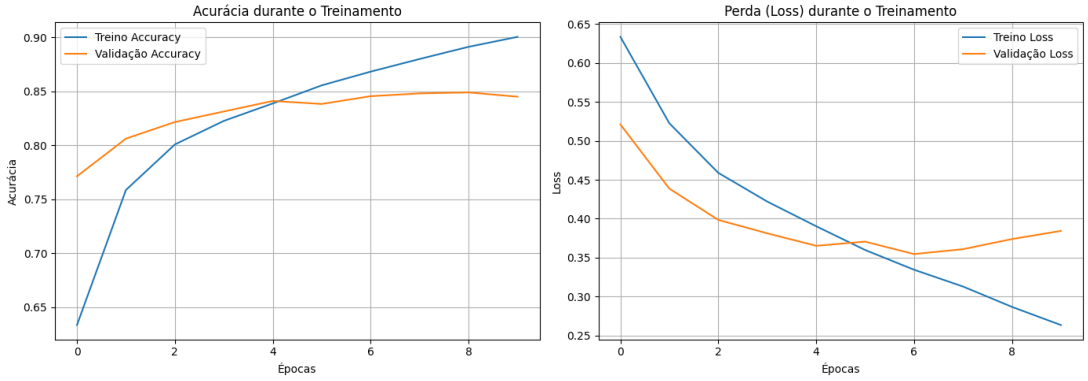


Figura 5.1: Gráfico de acurácia e perda durante o treinamento do modelo proposto.

Antes de avaliar os resultados e comparar com outros modelos, é importante analisar o gráfico 5.1, que mostra a evolução da acurácia e da perda (loss) durante o treinamento do modelo proposto (BERT + BiLSTM). Novamente, a linha azul representa o desempenho no conjunto de treinamento, enquanto a linha laranja representa o desempenho no conjunto de validação. Observa-se que a acurácia no conjunto de treinamento aumenta rapidamente nas primeiras épocas e depois começa a se estabilizar, indicando que o modelo está aprendendo os padrões dos dados de treinamento. A acurácia no conjunto de validação também aumenta de forma gradual, ao chegar na quarta época, as linhas cruzam-se, indicando que o modelo está começando a apresentar sinais de sobreajuste (*overfitting*). A perda (*loss*) diminui consistentemente no conjunto de treinamento, o que é esperado à medida que o modelo aprende. No entanto, após a quarta época, a perda no conjunto de validação começa a aumentar, reforçando a indicação de sobreajuste.

Essa análise sugere que o modelo proposto está aprendendo efetivamente os padrões dos dados, entretanto, ela começa a apresentar sinais de sobreajuste após a quarta época. Para mitigar esse problema, o *Early Stopping* foi implementado, interrompendo o treinamento quando o desempenho no conjunto de validação, pois não melhorou por 3 épocas consecutivas. Isso ajuda a garantir que o modelo mantenha um bom desempenho em dados não vistos, evitando o sobreajuste excessivo aos dados de treinamento.

A curva ROC (*Receiver Operating Characteristic*) do modelo proposto é apresentada na Figura 5.2. A curva ROC é uma ferramenta útil para avaliar o desempenho de modelos de modelagem binária, mostrando a relação entre a taxa de verdadeiros positivos (TPR) e a taxa de falsos positivos (FPR) em diferentes limiares de decisão. A área sob a curva ROC (AUC - *Area Under the Curve*) é uma métrica que quantifica a capacidade do modelo de diferenciar entre as classes positivas e negativas. Um AUC de 1 indica um modelo perfeito, enquanto um AUC de 0,5 indica um modelo que não tem capacidade discriminativa, equivalente a uma classificação aleatória.

A linha tracejada na Figura 5.2 representa a linha de base, que corresponde a um modelo sem capacidade discriminativa (AUC = 0,5). A curva ROC do modelo proposto está significativamente acima da linha de base, apresentando uma AUC de aproximadamente 0,93, indicando que o modelo tem uma excelente capacidade de distinguir entre títulos sarcásticos e não sarcásticos. Isso sugere que o modelo proposto é eficaz na tarefa de detecção de sarcasmo em títulos de notícias em português.

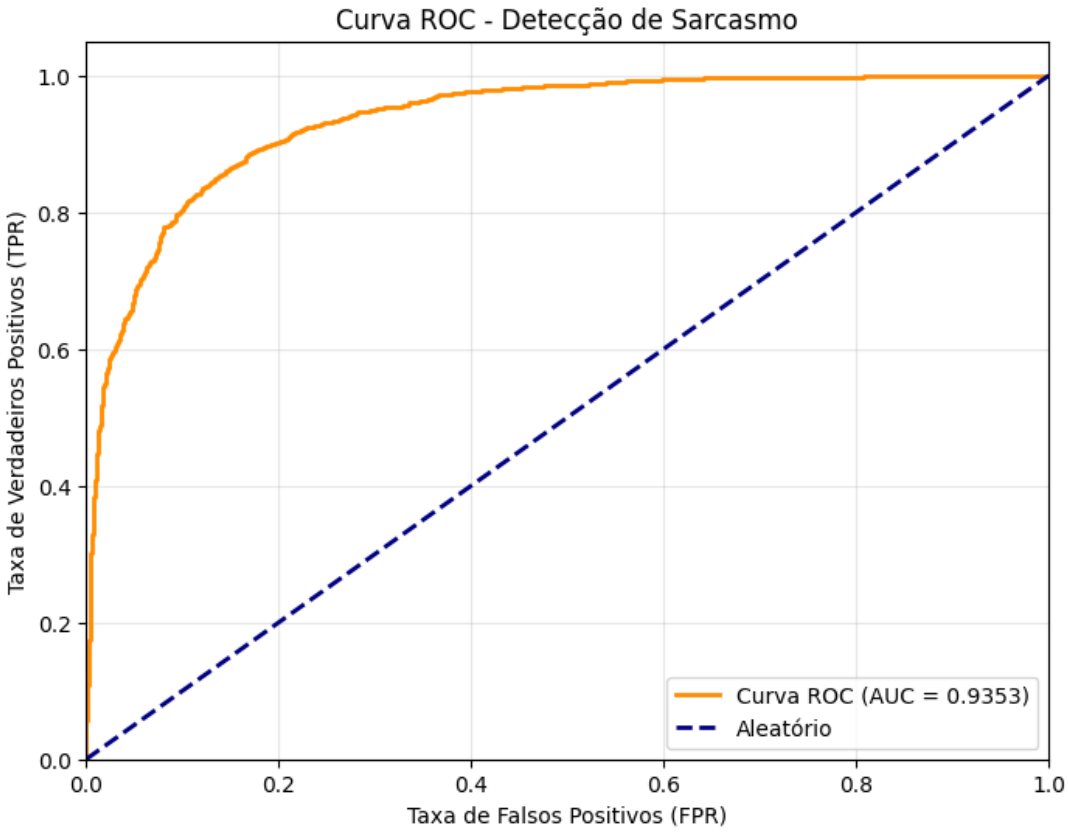


Figura 5.2: Curva ROC do modelo proposto.

A matriz de confusão do modelo proposto é apresentada na Figura 5.3. A matriz de confusão é uma ferramenta útil para visualizar o desempenho de um modelo de classificação, mostrando o número de verdadeiros positivos (TP), verdadeiros negativos (TN), falsos positivos (FP) e falsos negativos (FN). No contexto da detecção de sarcasmo em títulos de notícias, os verdadeiros positivos são os títulos sarcásticos corretamente classificados como sarcásticos, enquanto os verdadeiros negativos são os títulos não sarcásticos corretamente classificados como não sarcásticos. Os falsos positivos são os títulos não sarcásticos incorretamente classificados como sarcásticos, e os falsos negativos são os títulos sarcásticos incorretamente classificados como não sarcásticos.

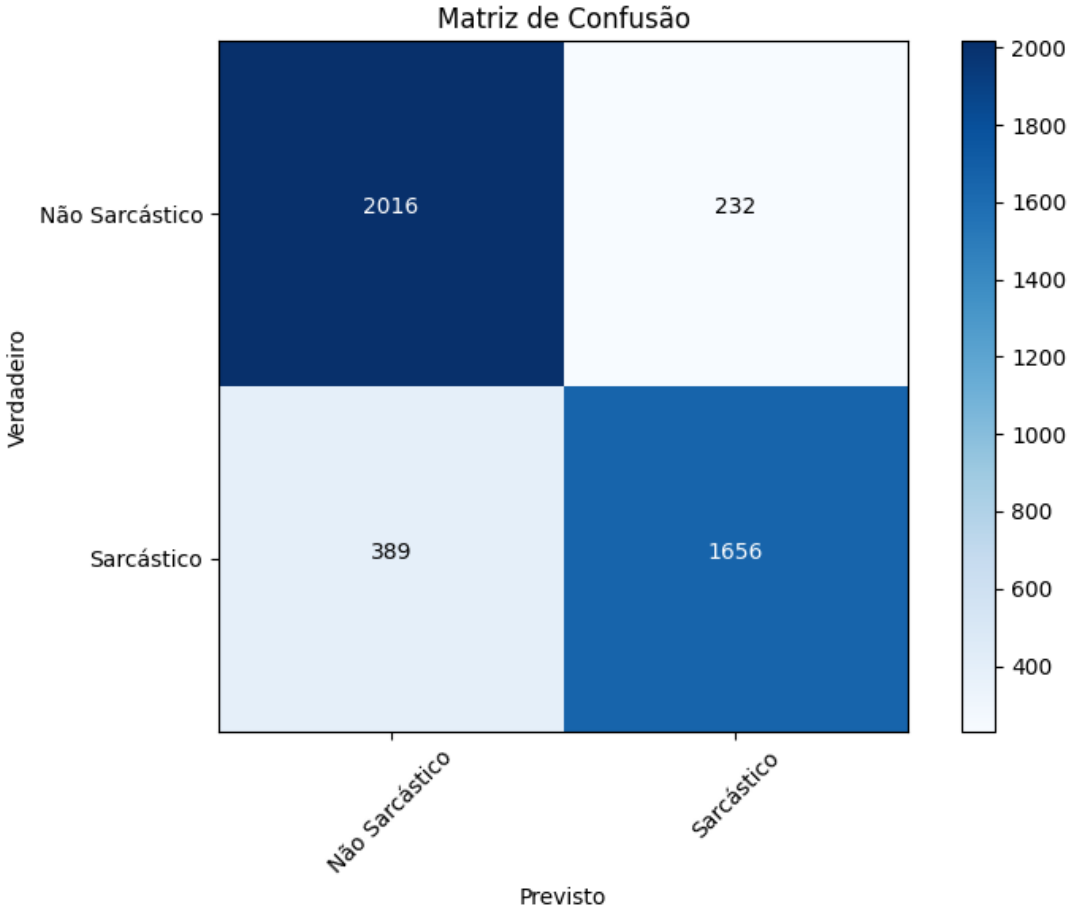


Figura 5.3: Matriz de Confusão do modelo proposto.

Observa-se na matriz de confusão que o modelo proposto classificou corretamente 1656 títulos sarcásticos (TP) e 2016 títulos não sarcásticos (TN). Houve 232 falsos positivos (FP) e 389 falsos negativos (FN). Esses resultados indicam que o modelo tem uma boa capacidade de classificação, embora haja espaço para melhorias, especialmente na redução do número de falsos negativos. A matriz também indica que o modelo é mais eficaz na identificação de títulos não sarcásticos do que na identificação de títulos sarcásticos.

Tabela 5.1: Resultados dos Experimentos de Detecção de Sarcasmo

	Base de Dados	Linguagem	Acurácia	Precisão	Recall	F1-Score
BiLSTM	Headlines	Português	83%	80%	82%	81%
BERT + BiLSTM	Headlines	Português	85%	87%	82%	85%
BiLSTM (Scola e Segura-Bedmar, 2021)	Headlines	Inglês	86%	85%	86%	86%
MHA-BiLSTM (Kumar et al., 2020)	Headlines	Inglês	•	72%	83%	77%
BiLSTM (Shrikhande et al., 2020)	Headlines	Inglês	86%	84%	86%	85%
BERT+BiLSTM (Nayak e Bolla, 2022)	Headlines	Inglês	89%	89%	89%	89%
CNN-BILSTM-Attention (Misra e Arora, 2023a)	Headlines	Inglês	90%	•	•	•
RoBERTa-CNN (Pawestri et al., 2024)	Headlines	Inglês	89%	88%	87%	87%
RoBERTa-BRNN (Pawestri et al., 2024)	Headlines	Inglês	65%	66%	64%	65%
RoBERTa+BiLSTM+MHA (Khan et al., 2025)	Headlines	Inglês	71%	69%	62%	65%

Os resultados apresentados na Tabela 5.1 indicam que o modelo proposto, BERT + BiLSTM, superou o modelo BiLSTM simples em todas as métricas avaliadas. A incorporação do BERT como extrator de características permitiu capturar melhor o contexto e as nuances da linguagem. No entanto, o modelo base foi superado por pouco, indicando que apesar das melhorias trazidas pelo BERT, o BiLSTM ainda é um modelo competitivo para a tarefa de detecção de sarcasmo.

Com relação ao modelo base (BiLSTM simples), a matriz de confusão é apresentada na Figura 5.4 e a curva ROC na Figura 5.5. A curva ROC do modelo base mostra uma AUC de aproximadamente 0,91, indicando uma boa capacidade de distinção entre as classes, embora ligeiramente inferior ao modelo proposto. A matriz de confusão revela que o modelo base classificou corretamente 1696 títulos sarcásticos (TP) e 1874 títulos não sarcásticos (TN), com 374 falsos positivos (FP) e 349 falsos negativos (FN). Esses resultados sugerem que o modelo base é eficaz, mas apresenta uma taxa maior de falsos positivos em comparação com o modelo proposto.

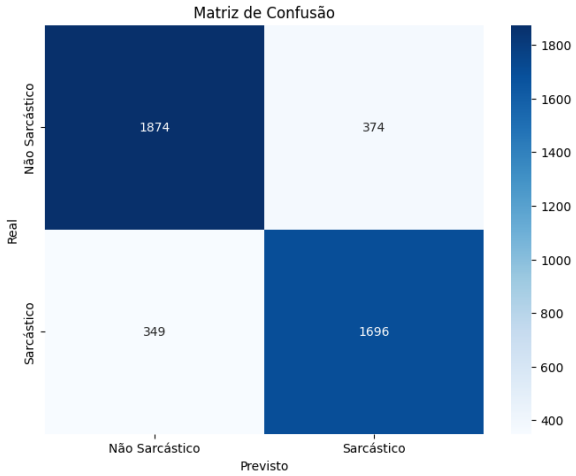


Figura 5.4: Matriz de Confusão do modelo base.

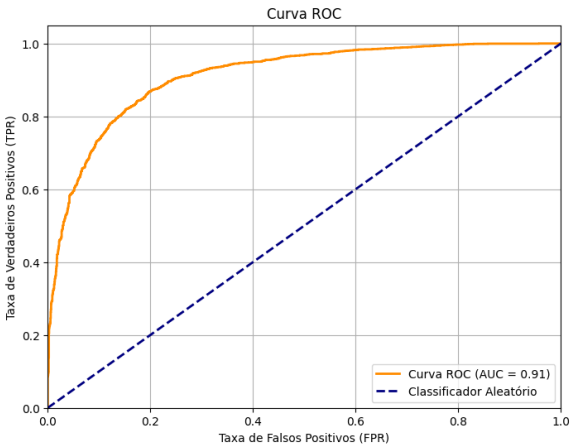


Figura 5.5: Curva ROC do modelo base.

Figura 5.6: Matriz de confusão e curva ROC do modelo base.

Se comparado com os trabalhos que utilizaram a mesma base de dados de notícias sarcásticas em inglês, o modelo proposto ficou atrás do modelo BERT + BiLSTM (Nayak e Bolla, 2022), que alcançou um F1-Score de 89% e do modelo RoBERTa-CNN (Pawestri et al., 2024). Mesmos modelos utilizando RoBERTa, um modelo mais recente e robusto que o BERT, como o RoBERTa + BiLSTM + MHA (Khan et al., 2025) e um dos modelos propostos por Pawestri et al. (2024) ficaram atrás do modelo proposto, indicando que a incorporação do BERT como extrator de características foi eficaz para a tarefa de detecção de sarcasmo em títulos de notícias em português.

Os resultados obtidos indicam que o modelo proposto é eficaz na detecção de sarcasmo em títulos de notícias em português, mas há espaço para melhorias. Futuras pesquisas podem explorar a incorporação de outras técnicas de pré-processamento, arquiteturas de modelos mais complexas e a utilização de bases de dados maiores e mais diversificadas para aprimorar ainda mais o desempenho na tarefa de detecção de sarcasmo.

5.3 CONSIDERAÇÕES

Neste capítulo, foram apresentados os experimentos realizados para avaliar o desempenho do modelo proposto na detecção de sarcasmo em títulos de notícias em português. Os resultados indicaram que a incorporação do BERT como extrator de características melhorou o desempenho em relação ao modelo BiLSTM simples. No entanto, o modelo proposto ainda ficou atrás de alguns modelos existentes na literatura. A comparação dos resultados deve ser feita com cautela, considerando as diferenças nas bases de dados e nas abordagens utilizadas. Futuras pesquisas

podem explorar novas técnicas e arquiteturas para aprimorar ainda mais o desempenho na tarefa de detecção de sarcasmo.

6 CONCLUSÃO E TRABALHOS FUTUROS

Muitos foram as dificuldades encontradas durante o desenvolvimento deste trabalho, desde a coleta dos dados, visto que é de difícil encontrar bases de dados anotadas para a tarefa de detecção de sarcasmo em português. A única base de dados que foi encontrada, era uma base privada, o que dificultou o desenvolvimento do trabalho, visto que não foi possível compartilhar a base de dados utilizada para treinamento e avaliação dos modelos. A alternativa encontrada foi a criação de uma base de dados própria, traduzindo uma base de dados em inglês para o português. Existem sites de notícias que possuiam manchetes sarcásticas, porém não seria possível utilizar essas manchetes, visto que elas eram protegidas por direitos autorais. A tradução não é algo ideal, visto que o sarcasmo pode ser culturalmente dependente, e uma frase sarcástica em inglês pode ser que não funcione da mesma forma em português. Com ajuda de modelos de linguagem, como o GPT-4.1 Mini, foi possível realizar a tradução de forma satisfatória, porém ainda assim, a base de dados criada não foi a mais linguisticamente rica possível.

No entanto, mesmo com as dificuldades encontradas, foi possível alcançar bons resultados com os modelos propostos. O modelo BiLSTM apresentou um *F1-score* de 81%, enquanto o modelo BERT-BiLSTM alcançou um *F1-score* de 85%, indicando que a incorporação do BERT trouxe melhorias no desempenho do modelo como esperado. Esses resultados são promissores, considerando as limitações da base de dados utilizada. É interessante notar que, embora o BERT tenha melhorado o desempenho, a diferença não foi tão significativa. Isso indica que o modelo BiLSTM já era bastante eficaz para a tarefa, isso somente com o texto como entrada, sem utilizar recursos linguísticos adicionais, apenas *embeddings* pré-treinados. Lembrado que embeddings nada mais são do que representações vetoriais das palavras, que capturam seu significado e contexto de uso. Visto que o BERT é um modelo pré-treinado que já possui conhecimento linguístico mais amplo e contextual, pensava-se que a diferença seria maior.

No entanto, os modelos propostos demonstraram ser eficazes para a tarefa de detecção de sarcasmo em textos em português, superando modelos tradicionais de aprendizado de máquina em várias métricas. Mostrando que para a tarefa de detecção de sarcasmo em manchetes em português, podem ser utilizados modelos baseados em redes neurais recorrentes.

Para trabalhos futuros, seria interessante explorar outras arquiteturas de modelos, como transformers mais avançados, ou mesmo modelos híbridos que combinem diferentes abordagens. Além disso, uma base de dados em português mais rica e diversificada poderia ser criada e utilizada para treinar e avaliar os modelos, o que poderia levar a melhorias significativas no desempenho e na generalização dos modelos desenvolvidos.

REFERÊNCIAS

(2024). Definição de sarcasmo - dicio, dicionário online de português. https://www.dicio.com.br. Acesso em: 26 dez. 2025.

Alpaydin, E. (2014). *Introduction to Machine Learning*. The MIT Press.

Baruah, A., Das, K., Barbhuiya, F. e Dey, K. (2020). Context-aware sarcasm detection using BERT. Em Klebanov, B. B., Shutova, E., Lichtenstein, P., Muresan, S., Wee, C., Feldman, A. e Ghosh, D., editores, *Proceedings of the Second Workshop on Figurative Language Processing*, páginas 83–87, Online. Association for Computational Linguistics.

de Medeiros Caseli, H. e das Graças Volpe Nunes, M. (2023). *Processamento de Linguagem Natural*. Brasileiras em PLN.

de Souza, F. C. (2020). Bertimbau = pretrained bert models for brazilian portuguese. Dissertação de Mestrado, Mestrado em Ciência da Computação - Universidade Federal de Campinas, Campinas - SP.

Devlin, J., Chang, M., Lee, K. e Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Goodfellow, I., Bengio, Y. e Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Grave, E., Bojanowski, P., Gupta, P., Joulin, A. e Mikolov, T. (2018). Learning word vectors for 157 languages. Em *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

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

Joulin, A., Grave, E., Bojanowski, P. e Mikolov, T. (2016). Bag of tricks for efficient text classification. *CoRR*, abs/1607.01759.

Khan, A., Majumdar, D. e Mondal, B. (2025). A hybrid transformer based model for sarcasm detection from news headlines. *Journal of Intelligent Information Systems*, 63:1339–1359.

Kumar, A., Narapareddy, V. T., Aditya Srikanth, V., Malapati, A. e Neti, L. B. M. (2020). Sarcasm detection using multi-head attention based bidirectional lstm. *IEEE Access*, 8:6388–6397.

Misra, R. e Arora, P. (2023a). Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18.

Misra, R. e Arora, P. (2023b). Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Nayak, D. e Bolla, B. (2022). *Efficient Deep Learning Methods for Sarcasm Detection of News Headlines*, páginas 371–382.

Pawestri, S., Murinto, M. e Auzan, M. (2024). Sarcasm detection: A comparative analysis of roberta-cnn vs roberta-rnn architectures. *Innovation in Research of Informatics (Innovatics)*, 6.

Rahma, A., Azab, S. S. e Mohammed, A. (2023). A comprehensive survey on arabic sarcasm detection: Approaches, challenges and future trends. *IEEE Access*, 11:18261–18280.

Razali, M. S., Halin, A. A., Ye, L., Doraisamy, S. e Norowi, N. M. (2021). Sarcasm detection using deep learning with contextual features. *IEEE Access*, 9:68609–68618.

Schuster, M. e Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Scola, E. e Segura-Bedmar, I. (2021). Sarcasm detection with bert. *Procesamiento del Lenguaje Natural*, 67(0):13–25.

Shrikhande, P., Setty, V. e Sahani, D. A. (2020). Sarcasm detection in newspaper headlines. Em *2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*, páginas 483–487.

Shu, X. (2024). Bert and roberta for sarcasm detection: Optimizing performance through advanced fine-tuning. *Applied and Computational Engineering*, 97:1–11.

Swami, S., Khandelwal, A., Singh, V., Akhtar, S. S. e Shrivastava, M. (2018). A corpus of english-hindi code-mixed tweets for sarcasm detection. *CoRR*, abs/1805.11869.

Tan, Y., Chow, C. O., Kanesan, J., Chuah, J. H. e Lim, Y. (2023). Sentiment analysis and sarcasm detection using deep multi-task learning. *Wireless Personal Communications*, 129:2213–2237.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. e Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.