# DESENVOLVIMENTO DE UM MODELO PREDITIVO PARA RECONHECIMENTO DE FRASES INDICATIVAS DE DEPRESSÃO EM REDES SOCIAIS

**SANTOS, Vinnicius[1]; FREITAS, Kenedy Antônio de[2]; TREVIZANO, Waldir Andrade[2]; PEREIRA, Ana Amélia de Souza[2];**

vinnicius109@gmakenedy
@unifagoc.edu.bril.com

[1] Graduando em Ciência da Computação pelo UNIFAGOC.
[2] Docente UNIFAGOC.

## RESUMO

*Este trabalho desenvolveu um protótipo de modelo preditivo baseado em inteligência artificial para identificar padrões de linguagem associados à depressão, com uma acurácia de 88,02%. Utilizando dados de redes sociais e uma rede neural LSTM, o modelo foi treinado e otimizado para detecção precoce da doença. Avaliado por métricas como precisão, sensibilidade e F1-score, mostrou-se uma ferramenta promissora para auxiliar profissionais de saúde no diagnóstico precoce.*

**Palavras-chave**: Depressão. Aprendizado de Máquina. Processamento de Linguagem Natural. Diagnóstico Precoce. Saúde Mental.

## INTRODUÇÃO

A depressão é um transtorno mental comum que além de comprometer a saúde mental, ela também tem um impacto significativo na qualidade de vida e na capacidade de integração social (OPAS, 2023). De acordo com a Organização Mundial da Saúde (OMS), estima-se que cerca de 300 milhões de pessoas em todo o mundo sofrem com essa enfermidade, sendo que uma em cada seis pessoas entre 10 e 19 anos enfrenta tal condição. Destaca-se que o suicídio, decorrente de transtornos mentais, é a quarta maior causa de morte entre jovens brasileiros com idades entre 15 e 29 anos, sendo a depressão a principal causa subjacente (OMS, 2023).No entanto, esse número pode ser ainda maior, já que na maioria das vezes ela não é diagnosticada ou tratada precocemente. Com a crescente banalização, o autodiagnóstico e a automedicação tornaram-se comuns, o que evidencia a urgência de alterar a forma como lidamos com esses pacientes (Pierri, 2021).

Durante a pandemia de COVID-19, os casos de depressão aumentaram em 25% (ONU, 2022). O isolamento social, a incerteza econômica e o medo da doença contribuíram para a banalização da depressão. Muitas vezes, os indivíduos acabam consumindo informações
incorretas sobre os sintomas, tratamentos e definições da depressão, o que pode levar a auto diagnósticos errôneos, automedicação inadequada e até mesmo agravamento dos sintomas (OMS, 2022).

Diante desse cenário desafiador, surge a necessidade de explorar soluções inovadoras que possam alcançar as pessoas no ambiente digital em que estão buscando assistência, sendo o desenvolvimento de um modelo inteligente alimentado por aprendizado de máquina uma ferramenta útil para apoiar esse

processo. Esse modelo pode desempenhar um papel auxiliar na detecção precoce da depressão, ao analisar padrões de linguagem e comportamento do usuário, identificando sinais precoces da condição e aprimorando sua precisão ao longo do tempo. Com isso, é possível reduzir o estigma em torno da depressão e garantir que mais pessoas recebam o apoio necessário para lidar com essa condição de maneira saudável (Constantino, 2023).

A busca indiscriminada por dados sobre depressão em fontes online não verificadas, como perfis em redes sociais, tem desencadeado um alerta entre profissionais da saúde sobre um novo tipo de paciente, demandando ações para estreitar o vínculo com o consultório médico (Padilla, 2023). Uma das ações é fornecer recursos online, com informações seguras e validadas.

Dessa forma, o desenvolvimento de um modelo preditivo alimentado por inteligência artificial, que seja capaz de assimilar através de um diálogo com o usuário frases indicativas de depressão, torna-se uma ferramenta útil para profissionais da saúde, em especial, no diagnóstico precoce da doença (Kaywan; Ahmed, 2023). Além da utilidade para os profissionais da saúde, os resultados dos testes podem auxiliar os usuários como indicativos de propensão ou não à doença.

O objetivo geral foi desenvolver e treinar uma rede neural multicamada capaz de analisar a linguagem escrita natural dos usuários, de forma que ela pudesse identificar sinais de propensão à depressão.

Como objetivos específicos:

- Realizou-se coletas e o pré-processamento de dados de textos de redes sociais disponíveis na plataforma *Kaggle*.
- Implementou-se e foi treinada uma rede neural LSTM.
- Avaliou-se a performance do modelo LSTM, observando as métricas de perda e acurácia (com validação), além de precisão, sensibilidade, especificidade e F1-score para uma visão abrangente do desempenho preditivo.
- Ajustou-se e foi otimizado o modelo para detecção precoce das frases indicativas de depressão.
- Comparou-se o modelo ajustado e otimizado com outros modelos preditivos de depressão disponíveis na plataforma Kaggle.

## REFERENCIAL TEÓRICO

### Aprendizado de Máquina

Aprendizado de Máquina é um campo de estudo dentro da inteligência artificial que permite que computadores adquiram a capacidade de aprender com dados e experiências, sem a necessidade de serem programados de forma explícita para cada tarefa. Essa área utiliza algoritmos e modelos matemáticos que analisam grandes volumes de dados, identificam padrões complexos e ajustam-se automaticamente para aprimorar seu desempenho ao longo do tempo (Samuel, 1959) .

Os métodos de aprendizado de máquina podem ser amplamente categorizados em três tipos principais: aprendizado supervisionado, em que o modelo é treinado com dados rotulados; aprendizado não supervisionado, no qual o sistema explora dados sem rótulos específicos para descobrir estruturas ou padrões ocultos; e aprendizado por reforço, que ensina o modelo a tomar decisões em um ambiente dinâmico, aprendendo com erros e sucessos acumulados (Equipe Predize,

2023).

As aplicações do aprendizado de máquina são vastas, desde diagnósticos médicos, onde auxilia na identificação de doenças com base em exames clínicos, até sistemas de recomendação, que analisam o comportamento do usuário para sugerir produtos, filmes e músicas. Outras áreas de aplicação incluem veículos autônomos, processamento de linguagem natural e análise de imagens, em que as máquinas conseguem reconhecer objetos e pessoas com alta precisão (Tableau, 2023).

**Rede Neural**

Uma rede neural é um modelo de aprendizado de máquina inspirado no funcionamento do cérebro humano, projetado para tomar decisões de maneira semelhante aos neurônios biológicos. Ela é composta por camadas de nós, ou neurônios artificiais, organizadas em três tipos principais: uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada nó está conectado a outros e possui um peso e um limiar específicos. Quando a saída de um nó ultrapassa o limiar, ele é ativado e transmite dados para a próxima camada; caso contrário, não há propagação de dados (IBM, 2024).

As redes neurais aprendem e aprimoram sua precisão por meio de dados de treinamento, o que as torna ferramentas poderosas em diversas áreas da ciência da computação e inteligência artificial. Após o processo de treinamento, elas conseguem realizar tarefas complexas, como reconhecimento de fala e imagem, com grande eficiência e rapidez. Em comparação com a análise manual por especialistas humanos, essas redes podem reduzir significativamente o tempo necessário para realizar essas tarefas, como exemplificado pelo algoritmo de pesquisa do Google (IBM, 2024).

***Rede Neural Recorrente (RNN)***

As Redes Neurais Recorrentes são uma classe de redes neurais projetadas para lidar com dados sequenciais, onde a ordem dos elementos é fundamental. A característica distintiva das RNNs é a presença de ciclos em sua arquitetura, permitindo que informações sejam transmitidas ao longo da sequência, preservando o contexto temporal (Almeida, 2023).

**Long Short-Term Memory (LSTM)**

A Long Short-Term Memory (LSTM) foi desenvolvida para superar as limitações das Redes Neurais Recorrentes (RNNs), especialmente o problema do desvanecimento e explosão de gradientes, que dificultavam o aprendizado de dependências temporais de longo prazo. A LSTM introduz unidades de memória que permitem a retenção de informações ao longo de períodos prolongados, o que a torna altamente eficaz em tarefas como tradução automática, previsão de séries temporais e análise de texto. Sua estrutura é composta por três portas: de entrada, de esquecimento e de saída. Essas portas controlam o fluxo de informações dentro da célula de memória, garantindo que as dependências temporais sejam preservadas de forma mais eficiente e eficaz (Almeida, 2023).

**Processamento de Linguagem Natural (PLN)**

O Processamento de Linguagem Natural (PLN) é uma área da ciência da computação que une linguística e aprendizado de máquina. Seu objetivo é fazer com que computadores consigam entender, interpretar e responder à linguagem humana de forma natural e no contexto certo. Usando análises linguísticas e algoritmos de aprendizado de máquina, o PLN ajuda as máquinas a detectar padrões, entender significados e captar nuances em textos e conversas. Aplicações do PLN são diversas: assistentes virtuais como Siri e Alexa, tradução automática, análise de sentimentos nas redes sociais, chatbots e sistemas de recomendação. Também é usado para reconhecer e gerar linguagem em tempo real, automação de atendimento ao cliente, revisão automática de texto, diagnóstico médico e análise de documentos legais (Lopes, 2024).

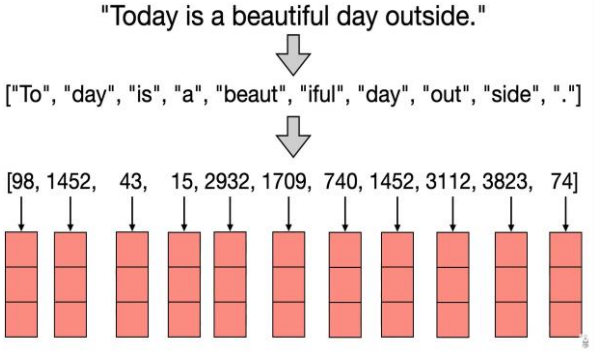
Para alcançar esse nível de interação, o PLN utiliza técnicas como a tokenização (divisão de textos em palavras ou frases), análise sintática (estrutura das frases), reconhecimento de entidades nomeadas (identificação de nomes de pessoas, lugares e organizações), entre outras. Esses processos possibilitam que as máquinas compreendam tanto a gramática quanto o contexto, interpretando ambiguidades e adaptando-se ao uso coloquial e formal da linguagem humana (Alves, 2013).

Dessa forma, o PLN representa um avanço significativo na construção de sistemas que interagem com seres humanos de forma mais intuitiva e eficaz, tornando a comunicação com a tecnologia mais acessível e personalizada (Alves, 2013).

**Tokenização**

A tokenização é um processo fundamental na análise de texto, que consiste na divisão de um texto em unidades menores denominadas tokens. Esse processo é realizado por um tokenizador, que identifica e segmenta as diferentes partes do texto, como palavras, sob palavras ou caracteres. Os tokens gerados são atribuídos com valores inteiros conhecidos como IDs de token, que servem para identificar de forma única cada token dentro de um vocabulário específico do tokenizador como mostra a Figura 1 abaixo (Alves, 2013).

**Figura 1 –** Demonstração do funcionamento do tokenizer

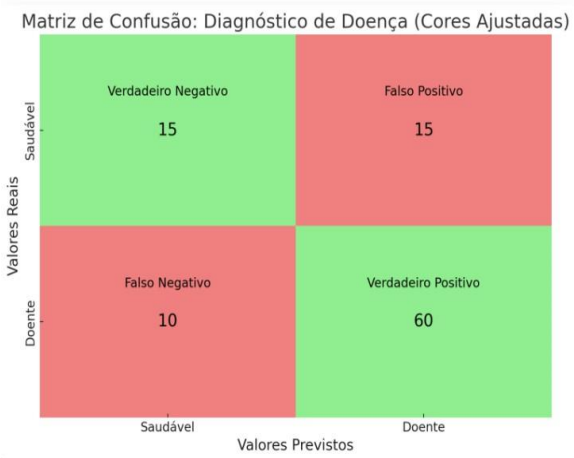



Fonte: DataMListic, 2024.

**Matriz de confusão**

A matriz de confusão é uma ferramenta essencial para avaliar a performance de modelos de classificação, composta por quatro quadrantes: Verdadeiro Positivo (VP), Falso Positivo (FP), Verdadeiro Negativo (VN) e Falso Negativo (FN), permitindo uma análise clara dos acertos e erros. Em problemas multiclasses, a matriz se expande, mas a interpretação dos componentes segue a mesma lógica (Pedro César Tebaldi Gomes, 2024). Na Figura 2, a classificação de pacientes quanto à condição de saúde mostra que o modelo acertou ao identificar 15 pacientes saudáveis (VN) e 60 doentes (VP), mas errou ao classificar 15 saudáveis como doentes (FP) e 10 doentes como saudáveis (FN) (Gomes, 2024).

**Figura 2 –** Exemplo da matriz de confusão em diagnósticos de doença




Fonte: Gomes, 2024

## MÉTODO DE DESENVOLVIMENTO

O desenvolvimento foi dividido em três etapas principais: Coleta e Pré-processamento dos Dados, em que realizamos a limpeza e transformação dos dados; Criação da Rede Neural, que envolveu a estruturação e o treinamento do modelo; Validação Avaliação e Comparação, na qual aplicamos métricas para verificar a precisão e eficácia do modelo final. Em seguida, apresenta-se a descrição detalhada de cada etapa.

**Coleta de Pré-processamento de Dados**

Na primeira etapa do projeto, os dados foram coletados da plataforma Kaggle, consistindo em dois conjuntos de dados: o primeiro contendo textos do Reddit[1] da comunidade r/depression e o segundo composto por postagens do Twitter[22].

Optou-se por utilizar um dataset em inglês devido a duas razões principais: a maior disponibilidade e a qualidade dos dados. De acordo com o levantamento bibliográfico para execução desse projeto, as línguas dominantes, como o inglês,

[1] Reddit Dataset: https://www.kaggle.com/datasets/infamouscoder/depression-reddit-cleaned/code
[2] Twitter Dataset: https://www.kaggle.com/datasets/mexwell/depressivenon-depressive-tweets-data/data

possuem vastas quantidades de dados de alta qualidade, o que é fundamental para a construção de modelos mais robustos em tarefas de Processamento de Linguagem Natural (PLN), como a detecção de sentimentos e tendências em textos (Bender, 2023).

Após a coleta, realizou-se a integração dos dois conjuntos de dados utilizando a biblioteca Pandas. Em seguida, efetuou-se a limpeza dos dados para garantir sua uniformidade, incluindo a remoção de caracteres especiais nos textos. Para a tokenização dos textos, utilizou-se a biblioteca Keras em conjunto com TensorFlow, por meio da função *text_to_sequences*, que converte as palavras em sequências de números, possibilitando o processamento eficiente dos dados textuais por modelos de aprendizado de máquina (Flama, 2024).

**Criação da Rede Neural LSTM**

Com a união e a limpeza finalizadas, iniciou-se a construção da rede neural utilizando as bibliotecas TensorFlow e Keras. A arquitetura da rede foi projetada para classificação binária de textos, com entrada de dados como sequências de palavras e saída com valores 0 ou 1, indicando a presença ou ausência de tendências à depressão. A rede incluiu uma camada de Embedding, que transformou os dados textuais em vetores densos de dimensão fixa, representando as palavras em um espaço contínuo e permitindo ao modelo aprender as relações semânticas entre elas (Brownlee, 2021). Em seguida, empregou-se uma camada LSTM (Long Short-Term Memory) com 128 neurônios, capaz de capturar dependências temporais e contextuais nas sequências de texto, aspecto essencial para tarefas de PLN, como análise de sentimentos e detecção de padrões de linguagem (Gomes, 2024).

Para evitar o *sobre ajuste do modelo*, foi aplicada uma camada de *Dropout* com uma taxa de 0.5 logo após a camada LSTM, que desativa aleatoriamente 50% dos neurônios durante o treinamento, ajudando o modelo a generalizar melhor (Yadav, 2022). A seguir, foi adicionada uma camada *Dense* com 64 neurônios e ativação *ReLU*, que permitiu a aprendizagem de padrões não lineares nos dados (Javid, Das, Skoglund, Chatterjee, 2021), seguida por outra camada de *Dropout* com taxa de 0.5, garantindo a prevenção de overfitting. Por fim, a camada de saída foi configurada com um único neurônio e ativação *Sigmóide*, que gera um valor entre 0 e 1, representando a probabilidade da classe positiva (ausência de tendências à depressão) ou negativa (presença de tendências à depressão).

Para a função de perda, utilizou-se *Binary Crossentropy (*GeeksForGeeks, 2024*)*, que é ideal para problemas de classificação binária, enquanto o *otimizador Adam* foi escolhido devido à sua capacidade de adaptação durante o treinamento (Brownlee, 2021). A taxa de aprendizado do otimizador foi configurada com o valor de 0.0001 (Beownlee, 2021), visando uma convergência estável e eficiente do modelo.

Em seguida, o conjunto de dados foi dividido em 80% para treino, 10% para teste e 10% para validação (Komesu, 2023), utilizando a função *train_test_split* da biblioteca scikit-learn. Para garantir uma divisão estratificada, preservando a distribuição dos rótulos nos dados de treino, teste e validação, foi utilizado o parâmetro *stratify=labels*, para assegurar que a amostra tenha a mesma proporção de rótulos em cada conjunto, e com *random_state* igual a *42* para garantir a reprodutibilidade dos resultados, permitindo que a divisão seja consistente a cada execução (Scikit Learn, 2024).

**Validação, Avaliação e Comparação do Modelo**

Na etapa de validação e avaliação do modelo, foram utilizadas técnicas e métricas essenciais para medir o desempenho do modelo. A matriz de confusão foi empregada para analisar a distribuição de verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos, o que permitiu uma análise detalhada da classificação do modelo (Gomes, 2024).

Além disso, observou-se a perda (loss) e a perda de validação (val_loss) por época para monitorar o aprendizado e o ajuste do modelo, verificando possíveis sinais de overfitting ou underfitting (Brownlee, 2019). A acurácia e a acurácia de validação (accuracy e val_accuracy) também foram analisadas para avaliar a taxa de acertos gerais do modelo a cada época.

As principais métricas de desempenho, como precisão, sensibilidade, especificidade e F1-score, complementam a avaliação, fornecendo uma visão abrangente sobre a capacidade preditiva e a robustez do modelo (Filho, 2023).

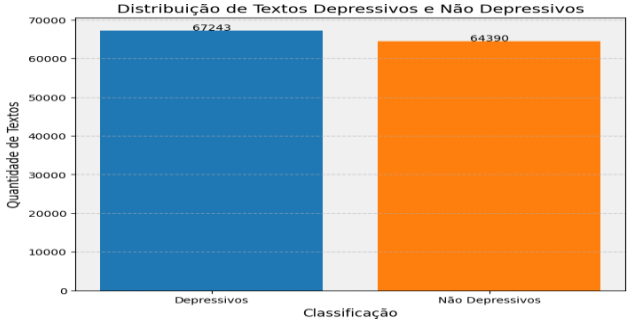
Após geradas as métricas de desempenho, comparou-se com outros modelos preditivos da plataforma *Kaggle, olhando para métricas como acurácia, precisão, recall e F1-score.*

## RESULTADO E DISCUSSÃO

Na primeira etapa do desenvolvimento do modelo preditivo, foram coletados 142.060 registros provenientes de dois conjuntos de dados, um do Reddit e outro do Twitter, contendo duas colunas: uma para o texto e outra para a classificação, sendo 0 para ausência e 1 para presença de tendências à depressão. Durante a preparação dos dados, foi identificado que 18 registros tinham informações ausentes e 10.412 eram duplicados, os quais foram removidos para garantir a qualidade e a integridade dos dados utilizados no treinamento do modelo, evitando vieses e distorções, o que contribui para a melhoria da precisão e generalização do modelo, além de prevenir problemas como sobreajuste. Após essa exclusão, realizou-se uma limpeza adicional no texto, removendo tags HTML, URLs, caracteres especiais e espaços extras, seguido pela conversão para letras minúsculas, tokenização e remoção de palavras de parada e pontuação. Durante essa limpeza, 32 registros ficaram vazios e também foram excluídos, resultando em um conjunto de dados com 92.825 palavras únicas (Data Science Team, 2022).

O conjunto final de dados, após o processo de limpeza, conteve 131.665 linhas e foi segmentado conforme apresentado no gráfico de barras da Figura 3 abaixo.

**Figura 3** – Visualização da divisão entre os dados




Fonte: dadods da pesquisa.

Depois de limpar os dados, eles foram divididos em 80% para treinamento e 20% para teste (Komesu, 2023). Isso resultou em 105.306 dados para treinamento e 26.327 para teste. A divisão foi feita de forma estratificada, usando o valor da coluna de rótulo, para garantir que a distribuição das classes fosse representativa em ambas as partes do conjunto de dados.

Em seguida, o texto passou por um pré-processamento. O número máximo de palavras únicas foi de 92.825, enquanto o número máximo de sequências foi definido como 28, com base no cálculo do comprimento das sequências (Vaswani, 2017). O número 28 foi calculado usando o percentil 95 do comprimento das sequências de palavras nos dados. Isso garante que a maioria das sequências tenha um comprimento adequado para o modelo, mantendo equilíbrio entre eficiência computacional e capacidade de capturar informações importantes.

Além disso, o *tokenizer* utilizado para a tokenização do texto foi salvo para uso posterior, permitindo que o mesmo processo de tokenização seja aplicado de maneira consistente em novos dados ou durante a fase de produção do modelo, garantindo a integridade e a reprodutibilidade dos resultados (Restack, 2024) .

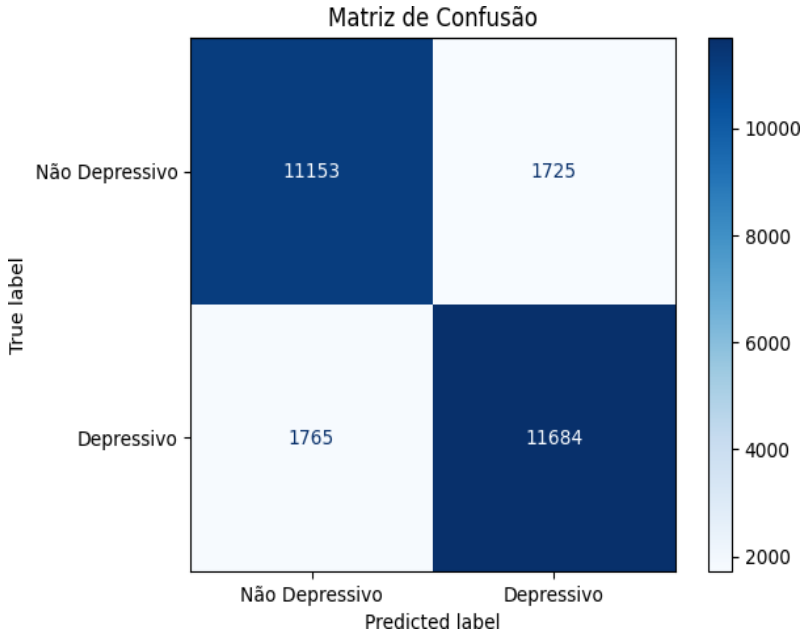
Após a coleta e pré-processamento dos dados, a construção da arquitetura do modelo foi iniciada usando técnicas para melhorar a eficiência e o desempenho do treinamento, todas oferecidas pela biblioteca Keras, do TensorFlow. Entre essas técnicas, destaca-se a redução dinâmica da taxa de aprendizado (ReduceLROnPlateau), que ajusta a taxa de aprendizado quando a métrica monitorada (val_loss) não apresenta melhorias após um número definido de épocas consecutivas (3, no caso do modelo), com um valor mínimo de taxa de aprendizado fixado em 0,000001 e fator de redução estabelecido em 0,2. Além disso, aplicou-se a técnica de checkpoint (ModelCheckpoint), que salva o modelo sempre que ele atinge a melhor acurácia de validação (val_accuracy), garantindo a preservação do melhor desempenho, e a técnica de interrupção antecipada (EarlyStopping), que interrompe o treinamento caso o val_loss não mostre melhorias após 10 épocas consecutivas, evitando o risco de overfitting. Essas estratégias, em conjunto, garantem um treinamento mais eficiente, resultando em um modelo com melhor desempenho, tanto em termos de acurácia quanto de generalização (Machine Learning Models, 2024).

Durante o treinamento do modelo inicial, observou-se um problema de sobreajuste (overfitting), caracterizado pela excessiva adaptação aos dados de treino e baixa capacidade de generalização para dados novos. Esse problema surgiu devido à simplicidade da arquitetura do modelo, que contava com poucas camadas e apresentava dificuldade em aprender representações complexas dos dados. Embora o modelo tenha apresentado bom desempenho no conjunto de treino, sua performance no conjunto de validação foi insatisfatória, evidenciando a necessidade de ajustes na arquitetura para reduzir o sobreajuste (Brownlee, 2020).

Para melhor avaliar o desempenho do modelo, uma matriz de confusão foi gerada, destacando uma quantidade considerável de falsos positivos e falsos negativos, o que reforçou a falta de generalização do modelo inicial. A Figura 4 ilustra essa matriz de confusão com as classificações realizadas e os erros cometidos. Já a Figura 5 apresenta as curvas de perda (loss e val_loss) e de acurácia (accuracy e

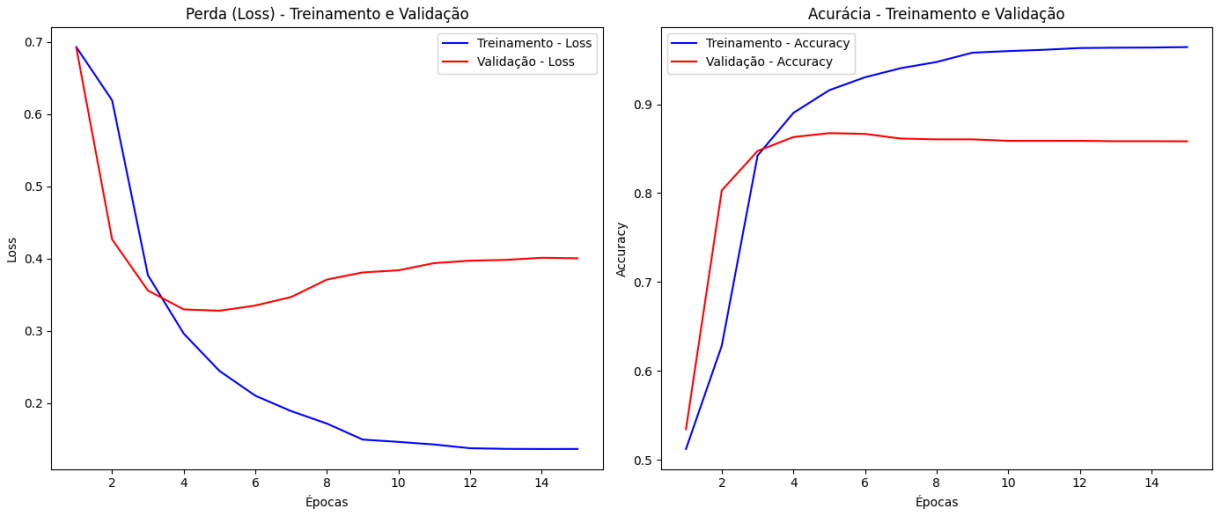
val_accuracy) para treino e validação, evidenciando a tendência ao sobreajuste com a progressão das épocas.

**Figura 4** – Matriz de confusão da primeira versão do modelo preditivo




Fonte: elaborada pelo autor.

**Figura 5** – Métricas da primeira versão do modelo preditivo




Fonte: elaborada pelo autor.

O modelo proposto no método do desenvolvimento era baseado em uma arquitetura simples, com uma camada LSTM, uma camada de *Dropout* e camadas densas subsequentes. No entanto, essa configuração resultou em um modelo que não era capaz de entregar uma generalização satisfatória. Por isso, foi necessário realizar modificações na arquitetura da rede neural (Simplescience, 2024).

Dessa forma, foram adicionadas novas camadas na arquitetura, com o objetivo de melhorar sua capacidade de representação e generalização. A estrutura original

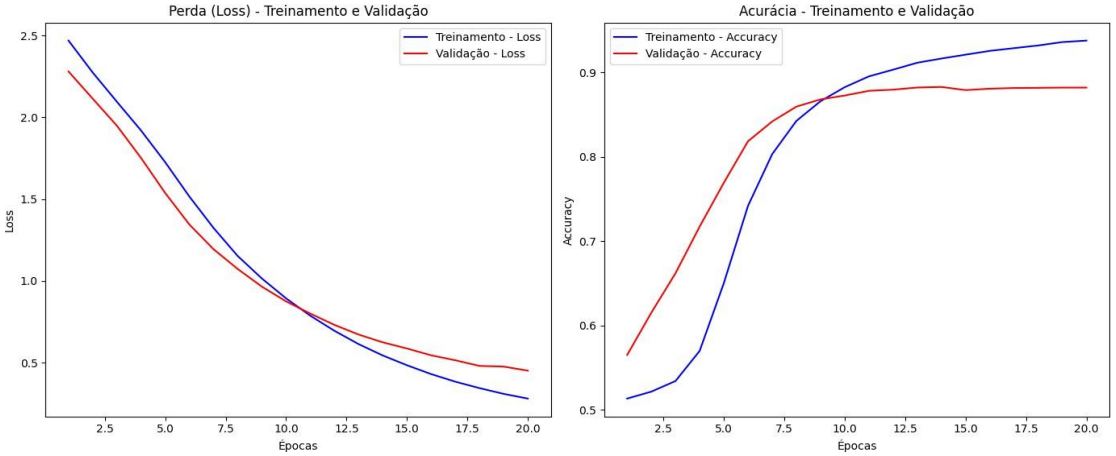
foi alterada para incluir múltiplas camadas LSTM bidirecionais, o que permitiu ao modelo processar as sequências em ambas as direções (frente e verso) com o objetivo de aproveitar a representação enriquecida dos dados (Anishnama, 2023). Cada camada bidirecional foi configurada com um grande número de unidades (256, 128 e 64) (Deep Learning Book, 2022).

Para combater o sobreajuste, o modelo passou a incluir camadas de normalização de lote (Batch Normalization) após cada camada LSTM, o que estabilizou a distribuição das ativações e acelerou o treinamento. Além disso, o Dropout foi inserido não apenas nas camadas mais profundas como também após a camada de entrada, com o uso de Spatial Dropout, e também após as camadas LSTM e densas, para ajudar a prevenir o sobreajuste e forçar o modelo a aprender representações mais robustas. O modelo também foi aprimorado com duas camadas densas de 128 e 64 unidades, configuradas com função de ativação ReLU. Cada camada densa usou regularização L2, o que ajudou a controlar a complexidade do modelo, incentivando pesos menores e reduzindo a sensibilidade do modelo a ruídos nos dados de treinamento (Lucas, 2024).

A taxa de aprendizado do otimizador Adam foi ajustada para um valor baixo (0.00005) para que as atualizações de peso fossem graduais, e nenhuma mudança violenta ocorresse no treinamento, contribuindo assim para uma convergência mais estável (ICHI, 2024).

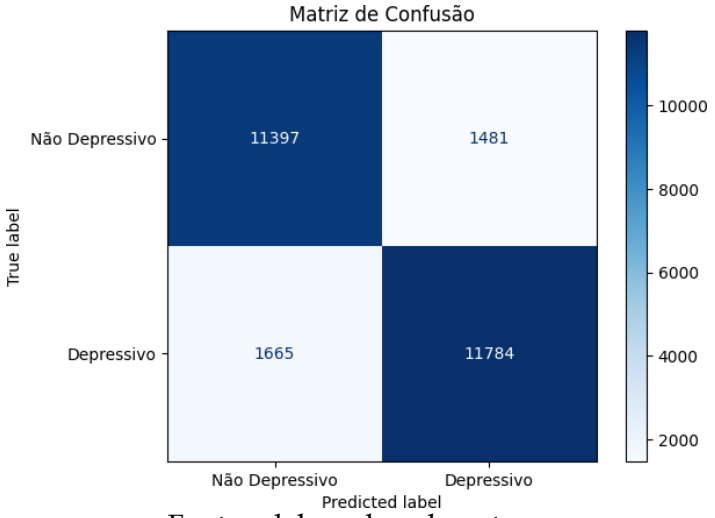
Após as modificações, o modelo apresentou melhorias significativas em termos de precisão e redução do sobreajuste, conforme demonstrado nas curvas de perda e acurácia da Figuras 6. Observou-se uma redução mais estável na perda e um aumento consistente na acurácia no conjunto de validação, evidenciando que o modelo foi capaz de aprimorar a generalização. A análise da matriz de confusão (Figura 7) também revela uma melhora na capacidade de classificação, com uma redução perceptível nos erros de previsão.

**Figura 6 –** Métricas da segunda versão do modelo




Fonte: elaborada pelo autor.

**Figura 7 –** Matriz de confusão da segunda versão do modelo preditivo




Fonte: elaborada pelo autor.

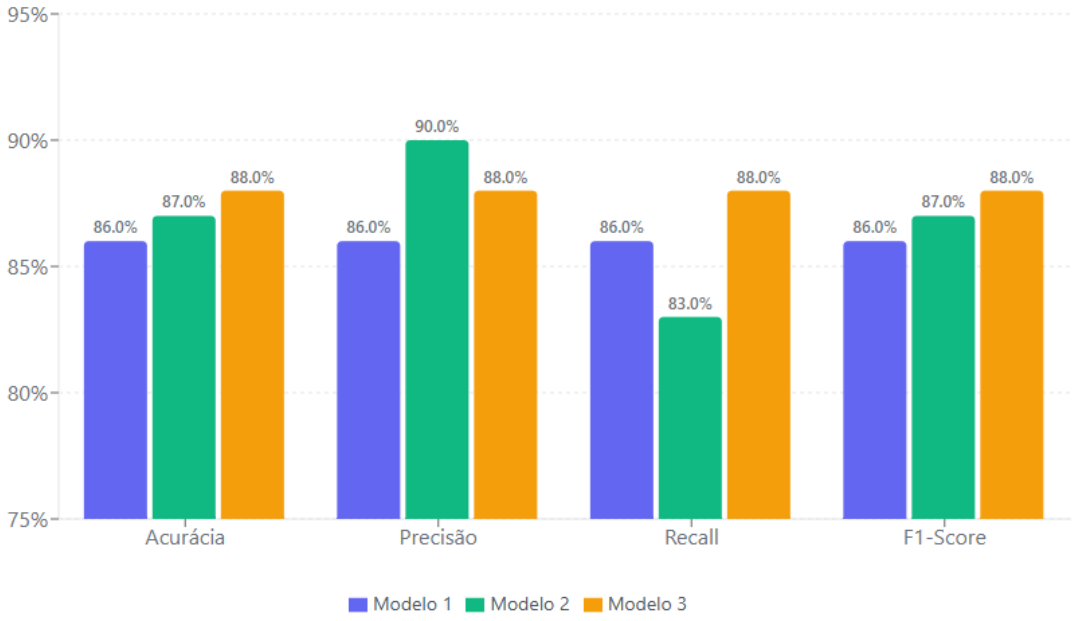
Essas melhorias tornaram o modelo mais robusto, equilibrando complexidade e capacidade de generalização. Além da acurácia, outras métricas foram utilizadas para avaliar o desempenho das duas versões. A acurácia com validação cruzada da primeira versão foi de 86,74%, enquanto a segunda versão alcançou 88,02%, indicando uma melhoria geral. A precisão, que representa a proporção de casos previstos como "depressivo" que realmente eram depressivos, aumentou de 87,12% para 88,45%. A sensibilidade (ou recall), que mede a capacidade de identificar corretamente os casos depressivos, foi de 86,88% na primeira versão e 88,25% na segunda. A especificidade, que reflete a precisão na identificação de casos "não depressivos", passou de 86,59% para 87,87%. Por fim, o F1-score, que é a média harmônica entre precisão e sensibilidade, subiu de 87,00% para 88,35%, confirmando uma melhoria consistente no desempenho da versão aprimorada do modelo (Campos, 2024).

**Comparação com outros modelos disponíveis**

Foi realizada uma comparação detalhada do desempenho entre três modelos: o Modelo 1[3], o Modelo 2[4] e o Modelo 3, desenvolvido neste projeto. A comparação considerou as principais métricas de avaliação: acurácia, precisão, recall e F1-Score (Figura 8).

[3] Disponível em: https://www.kaggle.com/code/grazynah/nlp-models-comparison-tensorflow
[4] Disponível em: https://www.kaggle.com/code/manas28/depressive-tweets-classification-pytorch-lstm

**Figura 8** – Comparação entre modelos disponíveis na plataforma *Kaggle*




Fonte: elaborada pelo autor.

A partir da análise dos resultados apresentados na Figura 8, observa-se que o Modelo 3, desenvolvido no projeto, alcançou a maior acurácia (88%), superando tanto o Modelo 1 (86%) quanto o Modelo 2 (87%). Em termos de precisão, manteve um desempenho robusto de 88%, posicionando-se entre o Modelo 2 (90%) e o Modelo 1 (86%). No quesito recall, destacou-se significativamente com 88%, superando consideravelmente o Modelo 2 (83%) e o Modelo 1 (86%). No F1-Score, métrica que representa o equilíbrio entre precisão e recall, o Modelo 3 também se sobressaiu com 88%, em comparação a 87% do Modelo 2 e 86% do Modelo 1. O diferencial mais notável do Modelo 3, desenvolvido no projeto, é sua consistência através de todas as métricas (88% em todas), indicando um desempenho mais equilibrado e confiável, sem apresentar quedas significativas em nenhum aspecto específico, como ocorre com o recall do Modelo 2 (83%).

## CONCLUSÃO

O trabalho teve como objetivo desenvolver um modelo inteligente, utilizando técnicas de inteligência artificial e processamento de linguagem natural para identificar padrões linguísticos associados à depressão e, assim, auxiliar no diagnóstico precoce de depressão. O objetivo principal de identificar de maneira precisa frases indicativas de depressão, foi abordado com sucesso através da criação e treinamento de uma rede neural LSTM multicamadas com acurácia de 88,02%, que se mostrou capaz de diferenciar padrões de linguagem correlacionados com sintomas depressivos em textos.

Ao longo do estudo, foi possível realizar uma coleta eficaz de dados, bem como seu pré-processamento, criando uma base consistente para o treinamento do modelo. A implementação e treinamento da rede neural permitiram resultados

satisfatórios na detecção precoce de indicadores de depressão, atendendo, portanto, ao objetivo proposto de contribuir como uma ferramenta adicional para profissionais de saúde. Além disso, o modelo ajustado foi comparado com outros modelos preditivos de depressão disponíveis na plataforma Kaggle, o que demonstrou seu potencial. A análise de performance do modelo revelou que ele pode oferecer uma precisão promissora, embora ainda dependa de ajustes contínuos para aumentar sua eficácia e se adaptar ao contexto dinâmico da linguagem usada nas redes sociais.

Deve-se considerar que, apesar dos avanços alcançados, o modelo não substitui a avaliação clínica, podendo atuar como um recurso complementar valioso. Como trabalhos futuros, recomenda-se expandir o treinamento com uma maior variedade de dados, incluindo diferentes culturas e contextos linguísticos, além de explorar modelos híbridos que integrem outras abordagens de inteligência artificial para refinar a acurácia e abrangência do diagnóstico. Outra linha futura relevante seria a criação de interfaces mais acessíveis para profissionais da saúde, facilitando a integração desse tipo de tecnologia no dia a dia do diagnóstico clínico e no acompanhamento de pacientes em risco de depressão.

## REFERÊNCIAS


ALMEIDA, Vitor. **Compreendendo Redes Neurais Recorrentes (RNNs) e LSTM**. Disponível em: https://blog.grancursosonline.com.br/compreendendo-redes-neurais-recorrentes-rnns-e-lstm/. Acesso em: jun. 2024.

ANISHNAMA. **Understanding bidirectional LSTM for sequential data processing**. Disponível em: https://medium.com/@anishnama20/understanding-bidirectional-lstm-for-sequential-data-processin g-b83d6283befc. Acesso em: out. 2024.

BARROSO LUCAS, Hudson. **Regularização em machine learning: técnicas e benefícios para modelos mais robustos**. Disponível em: https://iacomcafe.com.br/regularizacao-machine-learning-l1-l2-dropout-early-stopping-batch-normal ization/. Acesso em: out. 2024.

BENDER, E, M. *et al*. **On the dangers of stochastic parrots:** can language models be too big? Disponível em: https://dl.acm.org/doi/pdf/10.1145/3442188.3445922. Acesso em: out, 2024.

BROWNLEE, Jason. **Gentle introduction to the adam optimization algorithm for deep learning**. Disponível em: https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/. Acesso em: out. 2024.

BROWNLEE, Jason. **How to diagnose overfitting and underfitting of LSTM models**. Disponível em: https://machinelearningmastery.com/diagnose-overfitting-underfitting-lstm-models/. Acesso em: out. 2024.

BROWNLEE, Jason. **How to use learning curves to diagnose machine learning model performance**. Disponível em: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-perfor mance/. Acesso em: out. 2024.

BROWNLEE, Jason. **How to use word embedding layers for deep learning with keras**. Disponível em: https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/. Acesso em: out. 2024.

CAMPOS, Renata. **Matriz de confusão:** entenda para que serve esse tipo de tabela!. Disponível em: https://academiatech.blog.br/matriz-de-confusao/. Acesso em: out. 2024.

CONSTANTINO, Luciana. **Cientistas usam inteligência artificial e rede social para criar modelo que prevê ansiedade e depressão**. Disponível em: https://jornal.usp.br/ciencias/cientistas-usam-inteligencia-artificial-e-rede-social-para-criar-modelo-q%20ue-preve-ansiedade-e-depressao/. Acesso em: out. 2024.

DATA SCIENCE TEAM. **Limpeza de dados**. Disponível em: https://datascience.eu/pt/aprendizado- de-maquina/limpeza-de-dados/. Acesso em: out. 2024.

DATAMLISTIC**. Funcionamento do token.** Disponível em: https://youtu.be/hL4ZnAWSyuU. Acesso em: nov. 2024.

EQUIPE PREDIZE. **4 tipos de aprendizado de máquina que você precisa conhecer!**. Disponível em: https://predize.com/blog/tipos-de-aprendizado-de-maquina/. Acesso em: set. 2024.

FILHO, Mario. **As métricas mais populares para avaliar modelos de machine learning**. Disponível em: https://mariofilho.com/as-metricas-mais-populares-para-avaliar-modelos-de-machine-learning/. Acesso em: out. 2024.

FLAMA. **O que é Tokenização de texto**. Disponível em: https://flammadesign.com.br/glossario/o-que-e-tokenizacao-de-texto/. Acesso em: out. 2024.

GEEKS FOR GEEKS. **Binary cross entropy/log loss for binary classification**. Disponível em: https://www.geeksforgeeks.org/binary-cross-entropy-log-loss-for-binary-classification/. Acesso em: out. 2024.

GOMES, Carla. **Um processo independente de domínio para o povoamento automático de ontologias a partir de fontes textuais**. Disponível em: https://1library.org/document/zgxpml7q-processo-independente-dominio-povoamento-automatico- ontologias-partir-textuais.html. Acesso em: set. 2024.

GOMES, Pedro Cesar Tebaldi. **Matriz de confusão: Como utilizar para análise de modelos?**. Disponível em: https://www.datageeks.com.br/matriz-de-confusao/. Acesso em: out. 2024.

GOMES, Pedro César Tebaldi. **LSTM:** arquitetura de redes neurais long short term memory. Disponível em: https://www.datageeks.com.br/lstm/. Acesso em: out. 2024.

GOMES, Pedro César Tebald**i. Matriz de confusão:** como utilizar para análise de modelos? Disponível em: https://www.datageeks.com.br/matriz-de-confusao/. Acesso em: nov. 2024.

ICHI. **Estimando uma taxa de aprendizado ideal para uma rede neural profunda**. Disponível em: https://ichi.pro/pt/estimando-uma-taxa-de-aprendizado-ideal-para-uma-rede-neural-profunda-2267 18209371360. Acesso em: out. 2024.

IBM. **O que é uma rede neural?**. Disponível em: https://www.ibm.com/br-pt/topics/neural-networks. Acesso em: out. 2024.

KAYWAN, Payam; AHMED, Khandakar. **Early detection of depression using a conversational AI bot:** A non-clinical trial. Disponível em: https://journals.plos.org/plosone/article?id=10.1371/journal.pone. 0279743. Acesso em: out. 2024.

KOMESU. **Divisão de dados em treino, validação e teste para machine learning**. Disponível em: https://dkko.me/posts/treino-teste-validacao/. Acesso em: out. 2024.

LOPES, Renata. **Processamento de linguagem natural:** definição e exemplos de PLN. Disponível em: https://hub.asimov.academy/blog/processamento-linguagem-natural-pln/. Acesso em: out. 2024.

MACHINE LEARNING MODELS. **Overfitting in LSTM-based deep learning models**. Disponível em: https://machinelearningmodels.org/overfitting-in-lstm-based-deep-learning-models/. Acesso em: out. 2024.

ONU. **Depressive disorder (depression)**. Disponível em: https://www.who.int/news-room/fact-sheets/detail/depression/?gad_source=1&gclid=CjwKCAjwz42xBhB9EiwA48pT71UICtbzyQf-YcKEXf_EH77tfZ1Kh-CyMsLYKqba-oq09PS0MKqC0RoCKzQQAvD_BwE. Acesso em: abr. 2024.

OPAS. **Depressão**. Disponível em: https://www.paho.org/pt/topicos/depressa. Acesso em: abr. 2024.

PADILLA, José. **Quando as redes sociais aumentam o risco de depressão?**. Disponível em: https://amenteemaravilhosa.com.br/quando-as-redes-sociais-aumentam-o-risco-de-depressao/. Acesso em: out. 2024.

PIERRI, Vitória. **Banalização das doenças mentais dificulta diagnóstico e tratamento**. Disponível em: https://www.uol.com.br/vivabem/noticias/redacao/2021/02/21/banalizacao-das-doencas-mentais-di ficulta-diagnostico-e-tratamento.htm?cmpid=copiaecola. Acesso em: abr., 2024.

RESTACK. **Tokenizer save pretrained overview.** Disponível em: https://www.restack.io/p/tokenization-knowledge-tokenizer-save-pretrained-cat-ai. Acesso em: out. 2024.

SAMUEL, A. L. **Some studies in machine learning using the game of checkers**. Disponível em: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5392560. Acesso em: abr. 2024.

SCIKIT LEARN. **train_test_split**. Disponível em: https://scikit-learn.org/stable/modules/generated/ sklearn.model_selection.train_test_split.html. Acesso em: out. 2024.

SIMPLESCIENCE. **O que significa "Redes de Memória de Longo Prazo e Curto Prazo (LSTM)"?**. Disponível em: https://simplescience.ai/pt/redes-de-memoria-de-longo-prazo-e-curto-prazo-lstm--kgjzor. Acesso em: out. 2024.

TABLEAU. **A inteligência artificial no dia a dia em oito exemplos**. Disponível em: https://www.tableau.com/pt-br/learn/articles/ai/examples. Acesso em: out. 2024.

VASWANI. **Attention is all you need**. Disponível em: https://arxiv.org/abs/1706.03762. Acesso em: out. 2024.