

ANÁLISE DOS ALGORITMOS LSTM E GRU PARA A PREVISÃO DE VALOR DAS AÇÕES NA BOLSA DE VALORES BRASILEIRA

Jeferson Bonecher, Dalton Solano dos Reis – Orientador

Curso de Bacharel em Ciência da Computação
Departamento de Sistemas e Computação
Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brasil
jefbonecher@furb.br, dalton@furb.br

Resumo: Este estudo tem como objetivo analisar o desempenho dos algoritmos Long Short-Term Memory (LSTM) e Gated Recurrent Unit (GRU) na previsão de valores de ações na bolsa de valores brasileira. A metodologia envolveu a utilização de dados históricos de cinco empresas listadas na B3, aplicando técnicas de análise técnica e redes neurais recorrentes para prever a direção dos preços dos ativos. Os resultados indicaram que ambos os algoritmos são eficazes, com o modelo LSTM apresentando uma ligeira superioridade em termos de acurácia para algumas empresas. As métricas estatísticas utilizadas para avaliação incluíram Mean Squared Error (MSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) e Root Mean Squared Error (RMSE). Conclui-se que as redes neurais LSTM e GRU são ferramentas valiosas para a análise preditiva no mercado financeiro, embora apresentem limitações relacionadas à volatilidade do mercado e eventos inesperados.

Palavras-chave: Ações, Análise Técnica, Bolsa de Valores, Redes Neurais, LSTM, GRU, Variáveis.

1 INTRODUÇÃO

A bolsa de valores é um ambiente centralizado e especializado voltado para negociações no mercado de capitais, onde empresas e empreendimentos ofertam seus títulos, ações e instrumentos financeiros com o objetivo de atrair investidores e agentes interessados (ALMEIDA, 2022). Essa plataforma de negociações permite às organizações captarem recursos para financiar projetos e expandir atividades, enquanto investidores individuais e institucionais buscam diversificar seus portfólios e obter retornos financeiros através da compra e venda desses ativos (CAPITALRESEARCH, 2019).

Segundo Reis (2020), para um investidor analisar ações, existem duas abordagens principais: a análise fundamentalista e a análise técnica. A análise fundamentalista foca na qualidade e no potencial de crescimento de longo prazo de uma empresa, avaliando detalhadamente seus aspectos financeiros, como lucratividade e fluxo de caixa, além de fatores como governança corporativa e cenário macroeconômico. Em contraste, a análise técnica, utiliza padrões gráficos e indicadores de mercado, como médias móveis e tendências de preços, para identificar os momentos ideais de compra e venda, atraindo investidores interessados em aproveitar as variações de curto prazo do mercado. Cada método oferece perspectivas distintas, sendo a escolha dependente dos objetivos e do perfil de risco do investidor.

Neste trabalho é empregada a Análise Técnica, que segundo Pinto (2020), que é muito utilizada por investidores e profissionais do mercado financeiro para prever a direção dos preços de determinados ativos financeiros, com base na análise de gráficos que mostram a movimentação desses ativos ao longo do tempo. Por meio desta análise, busca-se identificar padrões e tendências que possam indicar movimentos futuros e assim, tomar decisões de "entrada" e "saída" com base nessas informações. A Análise Técnica é especialmente útil para operações de curto prazo na Bolsa de Valores, já que permite tomar decisões rápidas que podem resultar em lucro ou prejuízo em questão de minutos ou segundos, a depender da volatilidade do ativo em questão.

No campo da Análise Técnica, os algoritmos de Machine Learning, incluindo Redes Neurais (RN), têm ganhado cada vez mais espaço devido à sua capacidade de processar e analisar grandes volumes de dados em tempo real (XP EDUCATION, 2023). As Redes Neurais Recorrentes (RNR) são particularmente eficazes na análise de sequências temporais, como os dados de mercado financeiro, já que possuem a habilidade de capturar dependências temporais de longo prazo. Essa abordagem permite que investidores e *traders* identifiquem padrões e tendências nos mercados financeiros, auxiliando na tomada de decisões e potencialmente melhorando os resultados de suas operações (MATSUMOTO, 2019).

Nesse sentido, o presente trabalho tem como objetivo determinar qual conjunto de variáveis, aliado às redes neurais recorrentes Gated Recurrent Unit (GRU) e Long Short-Term Memory (LSTM), apresenta a melhor previsão de valores de ações na bolsa de valores brasileira. Para tanto, serão utilizados dados históricos de empresas envolvidas na B3 (bolsa de valores brasileira, sediada na cidade de São Paulo). Os objetivos específicos são: (i) prever a direção do movimento dos ativos, para os próximos 5 dias, das empresas Weg, Petrobras, Copel, Vale e Magazine Luiza, listada na bolsa de valores brasileira; (ii) avaliar o desempenho dos algoritmos GRU e LSTM utilizando métricas estatísticas como

acurácia, Mean Squared Error (MSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) e Root Mean Squared Error (RMSE)

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, são apresentados os principais assuntos que fundamentam e constroem as propostas deste estudo, envolvendo análise técnica e redes neurais, assim como os trabalhos correlatos que amparam as escolhas para a construção deste artefato de software.

2.1 ANÁLISE TÉCNICA

A Análise Técnica, também referida como Análise Gráfica, é uma técnica amplamente utilizada por *traders* para identificar oportunidades lucrativas no mercado de ações, especialmente em operações de curto e curtíssimo prazo, como Day Trade e Swing Trade (MODALMAIS, 2023). Esta metodologia concentra-se no estudo das variações de preços de ativos financeiros, principalmente por meio de gráficos, analisando padrões históricos de preços para antecipar movimentos futuros. Sua simplicidade e flexibilidade são características que a destacam, tornando-a uma abordagem popular e acessível para iniciantes no mundo do *trading* (PINHEIRO, 2020).

No centro da Análise Técnica estão os gráficos, que representam visualmente a evolução dos preços ao longo do tempo. Existem diferentes tipos de gráficos, como os de linhas, barras e *candlesticks*, cada um fornecendo informações específicas sobre os preços de abertura, fechamento, máximas e mínimas em determinados períodos (FERNANDES, HAMBERGER e VALLE, 2015). A interpretação desses padrões formados pelos gráficos é usada para identificar oportunidades de negociação.

Dentro desse contexto, surgem conceitos fundamentais como tendências, que são padrões de movimento dos preços (alta, baixa ou lateral), e suportes e resistências, níveis onde os preços frequentemente reagem, revertendo suas direções. Além disso, a Análise Técnica utiliza indicadores matemáticos, como médias móveis, RSI, MACD e Bandas de Bollinger, para auxiliar na identificação de sinais de compra e venda (GOULÃO, 2016).

A psicologia do mercado também deve ser considerada na Análise Técnica. Os padrões e movimentos observados nos gráficos refletem as emoções e comportamentos dos participantes do mercado, como ganância, medo, euforia e complacência. Compreender esses aspectos é essencial para interpretar corretamente os movimentos dos preços.

Essa técnica é amplamente aplicada em estratégias como Day Trade, que consiste em operações iniciadas e encerradas no mesmo dia para lucrar com pequenas variações de preço, e Swing Trade, que busca capturar movimentos maiores ao longo de dias ou semanas (MARTINS, 2012). Além disso, investidores de longo prazo também utilizam a Análise Técnica para identificar momentos oportunos de entrada em ativos.

Apesar de suas vantagens, a Análise Técnica possui limitações. A interpretação dos gráficos e indicadores pode variar entre *traders*, levando a diferentes decisões e resultados. Além disso, eventos inesperados ou notícias podem influenciar drasticamente os preços, desafiando as previsões feitas pela análise técnica. Portanto, é essencial complementar essa abordagem com outras formas de análise e estar ciente dos riscos inerentes ao mercado financeiro.

2.1.2 REDES NEURAIS

Mittal, em 2019, descreveu as redes neurais como sistemas algorítmicos que se assemelham ao funcionamento cerebral humano, com a capacidade de identificar padrões. Estas redes processam dados sensoriais, classificando ou agrupando informações não estruturadas, tais como imagens, sons, textos ou dados temporais, convertidos em vetores numéricos. Eles consistem em numerosos neurônios artificiais interligados, atuando em conjunto para resolver problemas específicos. A estrutura das redes neurais é caracterizada por múltiplas camadas de interferência que operam paralelamente, onde cada camada subsequente processa as saídas da camada anterior, semelhante ao modo como os neurônios humanos processam informações visuais.

Essa estrutura, proposta por autores como Rumelhart, Hinton e Williams (1986), em seu trabalho seminal sobre o aprendizado por retropropagação, permite que as Redes Neurais (RN) processem informações, capturando padrões e características dos dados de entrada. Cada neurônio em uma RN é conectado a neurônios das camadas anteriores e seguintes por meio de conexões que têm pesos associados. Esses pesos determinam a influência e a importância das entradas para o neurônio seguinte. Durante o treinamento, as RN ajustam esses pesos com base nos dados fornecidos, utilizando algoritmos de aprendizado, como a retropropagação do erro, para minimizar a diferença entre as previsões da rede e os resultados desejados.

O funcionamento das RN é baseado no processamento paralelo de informações através dos neurônios, onde cada neurônio aplica uma função de ativação para calcular a saída com base nas entradas e nos pesos associados a essas entradas. Essas funções podem variar, sendo algumas das mais comuns a função sigmoide, a função ReLU (Rectified

Linear Unit) e a função tangente hiperbólica, cada uma com suas características específicas em termos de comportamento e aplicabilidade (FALCÃO, 2019).

Um aspecto notável das RN é sua capacidade de aprender a partir dos dados. Esse aprendizado se dá por meio de repetidas exposições a conjuntos de dados, ajustando os pesos das conexões entre os neurônios para melhor se adaptar aos padrões presentes nos dados (FALCÃO, 2019). Esse processo de aprendizagem permite que as RNAs realizem uma ampla gama de tarefas, como reconhecimento de padrões, classificação, previsão e até mesmo a geração de novos dados, dependendo da arquitetura e do treinamento da rede.

2.1.2 REDES NEURAIS RECORRENTES

As Redes Neurais Recorrentes (RNR) são uma classe especial de redes neurais que têm a capacidade de processar sequências de dados, sendo aplicadas em tarefas que envolvem informações com dependências temporais, como análise de séries temporais, processamento de linguagem natural e reconhecimento de fala. Uma das principais características das RNR é a presença de *loops* em sua estrutura, permitindo que informações sejam persistidas e retroalimentadas na rede, tornando-as capazes de lidar com dados sequenciais (FREIRE; MENEZES JR; BARRETO, 2009).

A estrutura das RNR inclui neurônios que mantêm um estado interno, permitindo que informações de entradas anteriores influenciem nas saídas atuais. Isso as torna especialmente eficazes no processamento de sequências de dados de comprimento variável. No entanto, RNR tradicionais enfrentam desafios com problemas de desvanecimento ou explosão do gradiente, dificultando a retenção de informações relevantes em sequências muito longas (NELSON, 2017).

Para mitigar esse problema, foram desenvolvidas variantes mais avançadas de RNR, como Long Short-Term Memory (LSTM) e Gated Recurrent Unit (GRU) (ARUNKUMAR, 2021). Essas arquiteturas utilizam mecanismos de portas que regulam o fluxo de informações na rede, permitindo um melhor controle sobre o fluxo de gradientes e, consequentemente, a retenção de informações relevantes por períodos mais longos. As RNR têm aplicações diversas, incluindo tradução automática, geração de texto, análise de sentimentos em textos, previsão de séries temporais e reconhecimento de voz. Sua capacidade de lidar com contextos temporais complexos as torna extremamente úteis em problemas nos quais a ordem e o contexto dos dados são cruciais para a tomada de decisões ou inferências.

2.1.3 REDE LSTM

As Redes Neurais de Longa Memória Recorrente (LSTM - Long Short-Term Memory) são uma variação avançada das Redes Neurais Recorrentes (RNR). Elas foram projetadas para superar as limitações das RNR tradicionais no processamento de sequências de dados de comprimento longo, ao lidar de forma mais eficaz com o problema do gradiente desvanecente e explosivo. As LSTM foram introduzidas por Sepp Hochreiter e Jürgen Schmidhuber em 1997, e desde então têm sido amplamente adotadas em áreas como processamento de linguagem natural, análise de séries temporais e tarefas que envolvem dados sequenciais (HOCHREITER e SCHMIDHUBER, 1994).

A principal inovação das LSTM reside em sua arquitetura, que incorpora unidades de memória conhecidas como células. Essas células possuem mecanismos de portas, como a porta de entrada, a porta de esquecimento e a porta de saída, que regem o fluxo de informações na rede. Essas portas permitem que as LSTM aprendam quais informações reter ou descartar durante o processamento de sequências, tornando-as eficazes na captura de dependências de longo prazo nos dados.

A estrutura das LSTM permite a preservação de informações relevantes por períodos prolongados, tornando-as ideais para lidar com sequências de dados extensas. A capacidade de aprender dependências temporais complexas e reter informações significativas por longos períodos faz com que as LSTM sejam particularmente úteis em aplicações como geração de texto, tradução automática, reconhecimento de fala e previsão de séries temporais (PEREIRA, 2017).

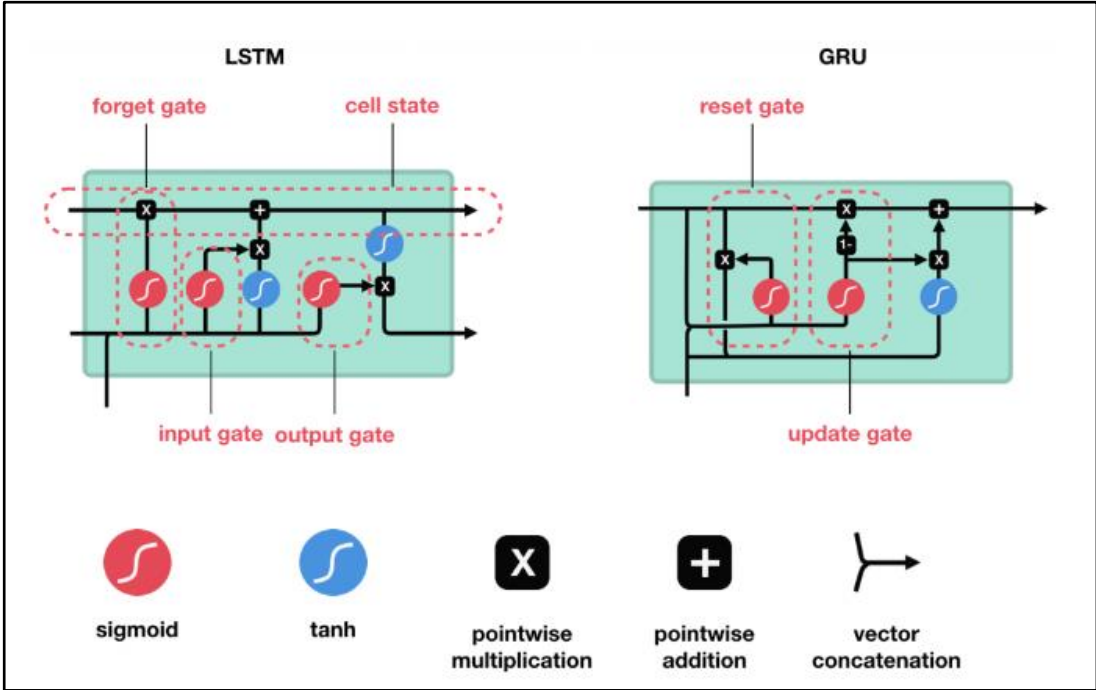
Essa arquitetura avançada de LSTM oferece uma solução eficiente para o desafio do gradiente desvanecente, permitindo que a rede mantenha um fluxo contínuo de informações ao longo das sequências. Isso proporciona maior estabilidade durante o treinamento e resulta em melhores capacidades de captura de padrões em dados sequenciais complexos.

2.1.4 REDE GRU

As Redes Neurais de Unidades Recorrentes com Portões Gated Recurrent Unit (GRU) são outra variante avançada das Redes Neurais Recorrentes (RNR), assim como as LSTM. Elas foram propostas por Cho et al. (2014) e se destacam por oferecer um mecanismo eficiente para lidar com dependências temporais em dados sequenciais, especialmente em problemas onde há restrições de recursos computacionais.

As GRU compartilham algumas similaridades com as LSTM ao utilizar mecanismos de portas para controlar o fluxo de informações dentro da rede. No entanto, as GRU são projetadas de forma mais simples, com uma estrutura mais compacta em comparação com as LSTM. Isso resulta em uma implementação computacionalmente mais leve e mais rápida em termos de treinamento e inferência. A implementação computacional da GRU pode ser observada na Figura 1.

Figura 1- LSTM vs GRU



Fonte: Deep Learning Book (2024).

Uma das características distintivas das GRU é a presença de duas portas principais: a porta de atualização e a porta de reinício. Essas portas permitem que as GRU aprendam quais informações são relevantes para manter e transmitir para as próximas etapas da sequência, ao mesmo tempo em que lidam de forma eficaz com problemas de desvanecimento e explosão do gradiente, tornando-as uma escolha viável para aplicações em que a eficiência computacional é crucial (MATSUMOTO, 2019).

2.2 TRABALHOS CORRELATOS

Nesta subseção são apresentados os trabalhos que correlacionam com o principal objetivo do presente trabalho. No Quadro 1 traz o trabalho desenvolvido por ROSSI (2019), em que foi criado um protótipo capaz de prever valores de ações através de Redes Neurais Artificiais. No Quadro 2 é apresentada uma aplicação que prevê o preço das ações por Redes Neurais Artificiais (TORRES, 2021). No Quadro 3 está uma aplicação com o objetivo de prever valores de ações através de RNRs (RIBEIRO; CORREA, 2021)

Quadro 1 – Trabalho Correlato 1

Referência	Rossi (2019).
Objetivos	Criar um protótipo capaz de prever tendências de preços de ações no mercado financeiro, utilizando Redes Neurais Artificiais (RNA) com foco na arquitetura LSTM.
Principais funcionalidades	Coleta de dados históricos das ações, incluindo preço de fechamento, preço de abertura, preço máximo, preço mínimo e volume negociado, a partir de fontes públicas. Realizar previsão dos próximos 5 preços utilizando os últimos 50 valores de fechamento. Avaliar o desempenho do modelo utilizando as métricas de erro MSE e MAE.
Ferramentas de desenvolvimento	Python, Framework Django, Biblioteca Bootstrap SB Admin 2, HTML, CSS, JavaScript, Long Short-Term Memory, MSE e MAE.
Resultados e conclusões	De acordo com Rossi (2019), os resultados mostraram que as ações RADL3 e WEGE3 obtiveram os melhores desempenhos. O modelo conseguiu prever corretamente a tendência de alta ou baixa nas cotações em 7 das 10 semanas avaliadas. Por outro lado, as ações VALE3 e BBDC4 apresentaram os piores resultados, com uma taxa de acerto de 30% e 40% na previsão, respectivamente. Essa diferença de desempenho pode ser atribuída à alta volatilidade desses ativos.

Fonte: elaborado pelo autor.

O trabalho de Rossi (2019) é um trabalho correlato relevante para o desenvolvimento do presente projeto, pois apresenta características semelhantes. Ambos os trabalhos utilizam Redes Neurais Recorrentes (RNR) para prever os preços de ações no mercado financeiro.

Quadro 2 – Trabalho Correlato 2

Referência	Torres (2021).
Objetivos	Prever o movimento das ações brasileiras utilizando Redes Neurais Artificiais (RNA) com recortes de tempo de 1, 2, 5, 10 e 20 anos. Comparar as arquiteturas LSTM e GRU para definir o melhor desempenho e a faixa temporal mais eficaz para análise.
Principais funcionalidades	Comparar o desempenho das arquiteturas LSTM e GRU, e determinar a melhor faixa temporal para análise.
Ferramentas de desenvolvimento	Python, Bibliotecas Keras, TensorFlow, Scikit Learn, Short-Term Memory, Gated Recurrent Unit, MSE e MAE.
Resultados e conclusões	Segundo Torres (2021) tanto a LSTM quanto a GRU obtiveram resultados parecidos as duas na faixa de 50%. Porém com tempos de análise diferentes. A LSTM foi melhor em dados até 2 anos a GRU foi melhor em dados acima de 10 anos. Como se pode observar nos gráficos a rede GRU teve o resultado melhor no índice Bovespa.

Fonte: elaborado pelo autor.

Torres (2021) utilizou Redes Neurais Artificiais (RNA) para prever o movimento das ações brasileiras, comparando as arquiteturas LSTM e GRU em recortes temporais de 1, 2, 5, 10 e 20 anos. O estudo concluiu que a LSTM teve melhor desempenho para períodos de até 2 anos, enquanto a GRU foi mais eficaz para análises acima de 10 anos. Este trabalho é relevante pois demonstra a eficácia dessas arquiteturas em diferentes horizontes temporais, corroborando a escolha das redes LSTM e GRU para previsões financeiras.

Quadro 3 – Trabalho Correlato 3

Referência	Ribeiro e Correa (2021).
Objetivos	Prever valores das ações no mercado brasileiro utilizando Redes Neurais Recorrentes (RNR) LSTM e GRU. Identificar a melhor rede e grupo de variáveis para previsões.
Principais funcionalidades	Utilização de métricas de análise técnica (médias móveis e bandas de Bollinger) combinadas com as redes neurais LSTM e GRU.
Ferramentas de desenvolvimento	Python, Biblioteca Yahoo Finance, Keras, Short-Term Memory, Gated Recurrent Unit, MinMaxScaler, MAPE, RMSE e U de Theil.
Resultados e conclusões	Os resultados do trabalho de Ribeiro e Correa (2021) indicaram que, de maneira geral, a inclusão de <i>features</i> adicionais aprimorou o desempenho das redes LSTM, resultando em médias mais baixas de erro percentual. Em relação às redes GRU, embora não tenha havido um cenário que se sobressaísse em todas as empresas analisadas, os resultados apontam que as <i>features</i> podem ser consideradas boas opções de métricas para previsões de valores de ações, devido ao baixo valor da média de erro percentual encontrado.

Fonte: elaborado pelo autor.

O trabalho de Ribeiro e Correa (2021) é um trabalho correlato relevante para a pesquisa proposta, pois aborda o mesmo problema de previsão de preços de ações utilizando redes neurais recorrentes. O trabalho apresenta resultados interessantes, mostrando que a inclusão de *features* adicionais pode melhorar o desempenho das redes LSTM.

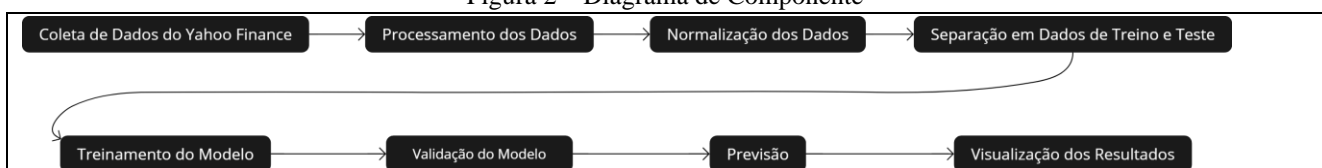
3 DESCRIÇÃO DO ARTEFATO DE SOFTWARE

Nesta seção, são descritos os aspectos mais relevantes para o desenvolvimento do artefato de software. Primeiramente, os Requisitos Funcionais (RFs) são apresentados no Quadro 4, enquanto os Requisitos Não Funcionais (RNFs) são exibidos no Quadro 5. Em seguida, são detalhados aspectos da implementação, incluindo a descrição dos dados, o pré-processamento realizado e as arquiteturas de redes neurais utilizadas.

3.1 ESPECIFICAÇÃO

A Figura 2 apresenta um diagrama de componente que mostra as principais etapas do processo de previsão de ações, desde a coleta de dados até a visualização dos resultados. Essa arquitetura é utilizada nos dois modelos de previsão LSTM e GRU.

Figura 2 – Diagrama de Componente



Fonte: elaborado pelo autor.

O Quadro 4 apresenta os Requisitos Funcionais (RFs) e o Quadro 5 relaciona os Requisitos Não Funcionais (RNFs).

Quadro 4 – Requisitos Funcionais

RF01 - O sistema deve gerar a previsão do valor de fechamento diário das ações
RF02 - O sistema deve gerar gráficos das previsões de valores das ações
RF03 - O sistema deve utilizar as métricas MSE, MAE, MAPE e RMSE para os dois algoritmos

Fonte: elaborado pelo autor.

Quadro 5 – Requisitos não Funcionais

RNF01 - O sistema deve ser implementado utilizando a linguagem Python
RNF02 - O sistema deve importar os dados históricos das ações através da biblioteca Yahoo Finance
RNF03 – O sistema deve utilizar a biblioteca Keras para desenvolvimento e treinamento dos modelos
RNF04 - O sistema deve ser capaz de prever os movimentos das ações nos próximos 5 dias
RNF05 - O sistema deve normalizar os dados obtidos
RNF06 - O sistema deve utilizar a biblioteca TensorFlow

Fonte: elaborado pelo autor.

3.2 IMPLEMENTAÇÃO

Para o desenvolvimento deste artefato de software, utilizou-se a linguagem de programação Python, sendo desenvolvido na IDE Google Colab, que oferece um ambiente baseado em nuvem com suporte para execução de códigos Python e integração com bibliotecas de aprendizado de máquina (SANTODIGITAL, 2023). Foi utilizada a base de dados disponibilizada pelo Yahoo Finance (Yfinance), na qual foram obtidos dados históricos de cinco anos de cinco empresas listadas na bolsa brasileira (B3) com frequência diária. Os dados incluem os valores de abertura, máxima, mínima e de fechamento. No Quadro 6, são apresentados os nomes das empresas e seus respectivos *tickers*.

Quadro 6 – Empresa Ticker

Empresa	Ticker
Weg	WEGE3
Petrobras	PETR4
Copel	CPLE6
Vale	VALE3
Magazine Luiza	MGLU3

Fonte: elaborado pelo autor.

O processamento dos dados envolveu o cálculo das Bandas de Bollinger, uma técnica de análise técnica que utiliza a média móvel simples (MM) e o desvio padrão para determinar as bandas superior (BB_sup) e inferior (BB_inf), indicando a volatilidade do ativo e potenciais condições de sobrecompra ou sobrevenda (NADAI, 2024). A função utilizada para esse cálculo aplicou uma janela de 20 dias. No Quadro 7 é apresentada a função para calcular as Bandas de Bollinger. Também foram utilizados o valor de fechamento e a média dos preços diários.

Quadro 7 – Função Calcular Bandas de Bollinger

```
# Função para calcular bandas de Bollinger
def calcular_bandas_bollinger(df, n=20):
    df['MM'] = df['Close'].rolling(window=n).mean()
    df['BB_sup'] = df['MM'] + 2 * df['Close'].rolling(window=n).std()
    df['BB_inf'] = df['MM'] - 2 * df['Close'].rolling(window=n).std()
    df = df.dropna() # Remover valores NaN
    return df
```

Fonte: elaborado pelo autor.

Após o processamento dos dados, eles foram divididos em dois conjuntos: um de treino e outro de teste, onde 80% dos dados foram alocados para treino e 20% para teste. Um *buffer* de 200 dias foi adicionado ao conjunto de teste para assegurar a continuidade das séries temporais e preservar a precisão dos cálculos subsequentes. Como os dados podem apresentar grande variação de valores ao longo do treinamento dos modelos, foi necessário normalizá-los. Para isso, utilizou-se a técnica MinMaxScaler, que transforma os dados para um intervalo entre 0 e 1 (SILVA, 2023). Em seguida é feito o pré-processamento para estruturar os dados em janelas deslizantes de 200 dias. No Quadro 8 são apresentadas as etapas de normalização e pré-processamento dos dados. Este pré-processamento assegura que os dados estejam adequadamente preparados para alimentar os modelos de previsão.

Quadro 8 – Normalização e Pré-Processamento dos Dados

```
# Normalização dos dados
scaler_close = MinMaxScaler(feature_range=(0,1))
treino_close_norm = scaler_close.fit_transform(treino_data['Preço de Fechamento'].values.reshape(-1,1))
teste_close_norm = scaler_close.transform(teste_data['Preço de Fechamento'].values.reshape(-1,1))

scaler_media = MinMaxScaler(feature_range=(0,1))
treino_media_norm = scaler_media.fit_transform(treino_data['Média'].values.reshape(-1,1))
teste_media_norm = scaler_media.transform(teste_data['Média'].values.reshape(-1,1))

scaler_bb_sup = MinMaxScaler(feature_range=(0,1))
treino_bb_sup_norm = scaler_bb_sup.fit_transform(treino_data['BB_sup'].values.reshape(-1,1))
teste_bb_sup_norm = scaler_bb_sup.transform(teste_data['BB_sup'].values.reshape(-1,1))

scaler_bb_inf = MinMaxScaler(feature_range=(0,1))
treino_bb_inf_norm = scaler_bb_inf.fit_transform(treino_data['BB_inf'].values.reshape(-1,1))
teste_bb_inf_norm = scaler_bb_inf.transform(teste_data['BB_inf'].values.reshape(-1,1))

# Pre-processamento para o treino
X_treino_close, X_treino_media, X_treino_bb, y_treino = [], [], [], []
for i in range(periodo_dias, len(treino_close_norm)):
    X_treino_close.append(treino_close_norm[i-periodo_dias:i])
    X_treino_media.append(treino_media_norm[i-periodo_dias:i])
    X_treino_bb.append(np.hstack((treino_bb_sup_norm[i-periodo_dias:i], treino_bb_inf_norm[i-periodo_dias:i])))
    y_treino.append(treino_close_norm[i])

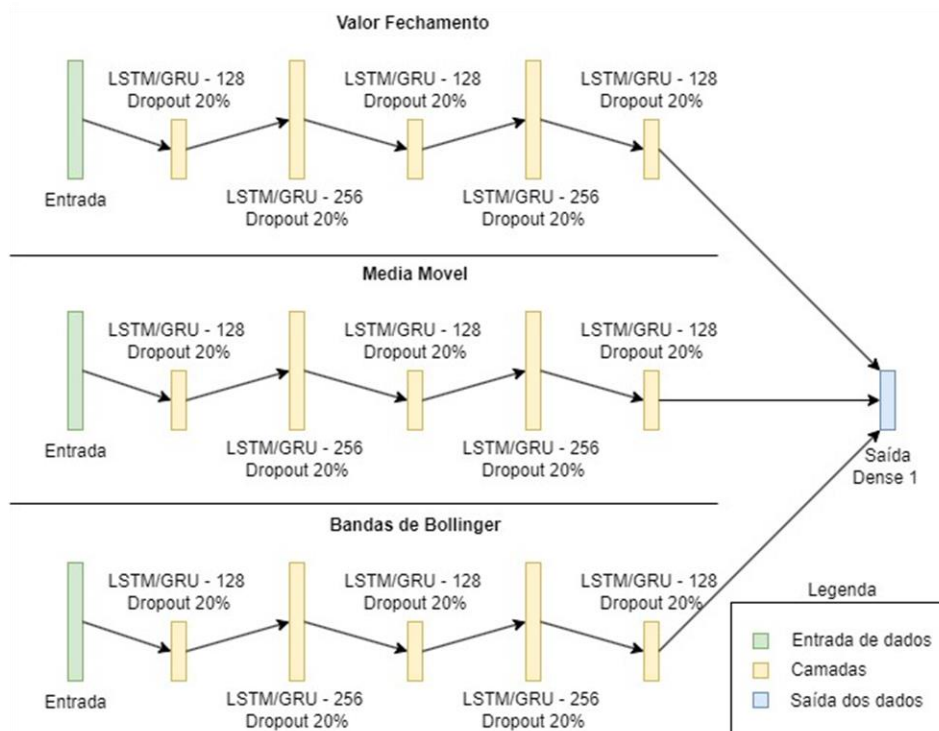
# Pre-processamento para o teste
X_teste_close, X_teste_media, X_teste_bb = [], [], []
for i in range(periodo_dias, len(teste_close_norm)):
    X_teste_close.append(teste_close_norm[i-periodo_dias:i])
    X_teste_media.append(teste_media_norm[i-periodo_dias:i])
    X_teste_bb.append(np.hstack((teste_bb_sup_norm[i-periodo_dias:i], teste_bb_inf_norm[i-periodo_dias:i])))

# Converter tudo para matriz numpy
X_treino_close, X_treino_media, X_treino_bb, y_treino = np.array(X_treino_close), np.array(X_treino_media), np.array(X_treino_bb), np.array(y_treino)
X_teste_close, X_teste_media, X_teste_bb = np.array(X_teste_close), np.array(X_teste_media), np.array(X_teste_bb)
```

Fonte: elaborado pelo autor.

Após esses processos, são criados os módulos preditores utilizando as redes neurais LSTM e GRU. Ambas utilizam os mesmos parâmetros e possuem 5 camadas, variando de 128 a 256 neurônios, conforme ilustrado na Figura 3. A primeira camada contém 128 neurônios, seguida por uma camada de Dropout de 20% para prevenir overfitting. Em seguida, há uma camada com 256 neurônios e 20% de Dropout, seguida por outra camada de 128 neurônios e 20% de Dropout. A quarta camada possui 256 neurônios com 20% de Dropout, e, finalmente, a quinta camada tem 128 neurônios, também com 20% de Dropout. Após esse processo ser realizado pelas três variáveis de entradas os valores são concatenadas e passam por uma camada densa para produzir a previsão final do preço de fechamento.

Figura 3 – Modelo de Treinamento



Fonte: elaborado pelo autor.

Os modelos são então compilados utilizando o otimizador Adam e a função de perda MSE. As métricas utilizadas para avaliação incluem MAE e MAPE. Para melhorar o treinamento, são utilizados callbacks como EarlyStopping, que interrompe o treinamento se a perda de validação não melhorar após 10 épocas, e o ModelCheckpoint, que salva o modelo apenas quando a perda de validação é a melhor.

Os modelos são treinados utilizando os dados de treino, com 20% dos dados reservados para validação. O treinamento é realizado por 50 épocas com um tamanho de batch de 64.

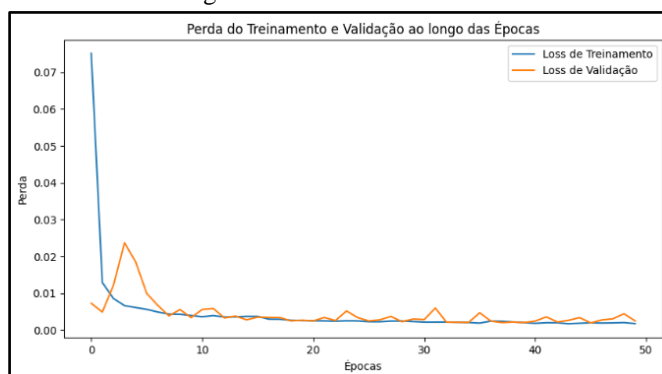
4 RESULTADOS

Para avaliar o desempenho dos modelos LSTM e GRU foram utilizadas métricas de erro, como o MAE, que calcula a média dos valores absolutos das diferenças entre previsões e valores reais; o MSE, que dá mais peso aos maiores erros por meio da média das diferenças quadráticas; e o RMSE, que é a raiz quadrada do MSE e retorna o erro à unidade de medida do modelo. Além disso, o MAPE é uma métrica útil em previsão de demanda, mostrando a porcentagem média de erro em relação aos valores reais e calculando a soma dos valores absolutos das diferenças pelo valor absoluto dos valores reais, multiplicando o resultado por 100 (OLIVEIRA JÚNIOR, 2021).

A métrica MSE foi utilizada em dois momentos. O primeiro MSE, de treinamento, refere-se ao erro calculado no conjunto de treinamento, medido pela função de perda, que calcula a discrepância entre as previsões do modelo e os valores reais, representando a média desses erros ao longo das épocas. Já o segundo MSE, de validação, é o erro calculado no conjunto de validação, que utiliza uma parte dos dados reservada para validação, neste caso, 20%. Embora o modelo não seja treinado com esses dados, ele faz previsões com eles após cada época, fornecendo uma medida de quão bem o modelo está generalizando para dados não vistos durante o treinamento.

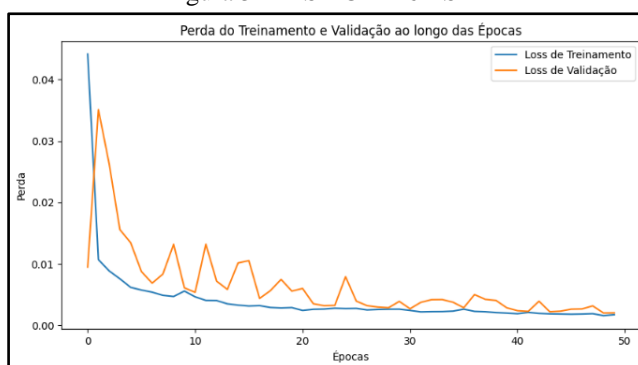
Como podemos ver na Figura 4, a partir da décima época, o gráfico se tornou linear. Já na Figura 5, a linha de validação se normalizou próximo das cinquenta épocas.

Figura 4 – MSE CPLE6 GRU



Fonte: elaborado pelo autor.

Figura 5 – MSE CPLE6 LSTM



Fonte: elaborado pelo autor.

Como podemos ver nos Quadro 9 e Quadro 10, os resultados dos dois modelos são muito parecidos, sendo a maior diferença observada com a PETR4.SA. No modelo GRU, obteve-se um MSE de treinamento de 0.0021 e de validação de 0.0013, enquanto na LSTM os valores foram de 0.0031 no treinamento e 0.0042 na validação. A melhor acurácia ficou com o modelo LSTM para a empresa VALE3, com 72%, e a pior ficou com a MGLU3, com 21%, tanto na LSTM quanto na GRU.

Quadro 9 – Métricas GRU

Empresa	MSE Treinamento	MSE Validação	MAE	MAPE	RMSE	Acurácia
MGLU3.SA	0.0015	0.0001	2.1954	0.1225	0.0112	21%
CPLE6.SA	0.0017	0.0025	0.2356	0.0251	0.0495	65%
VALE3.SA	0.0018	0.0008	1.7581	0.0266	0.0288	69%
PETR4.SA	0.0021	0.0013	0.8104	0.0219	0.0360	52%
ITSA4.SA	0.0012	0.0006	0.1468	0.0151	0.0244	67%

Fonte: elaborado pelo autor.

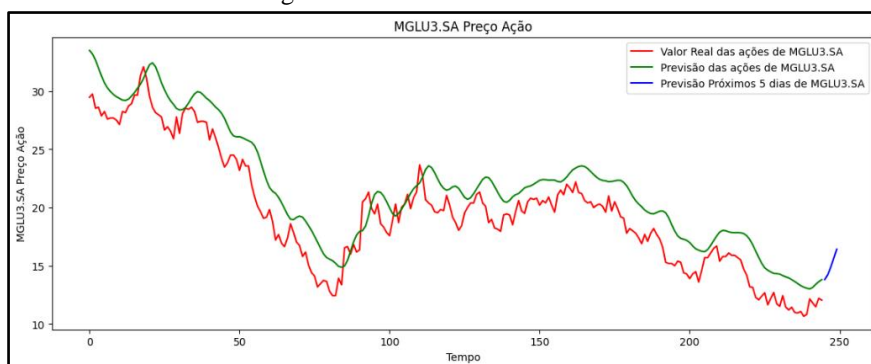
Quadro 10 – Métricas LSTM

Empresa	MSE Treinamento	MSE Validação	MAE	MAPE	RMSE	Acurácia
MGLU3.SA	0.0015	0.0003	1.5980	0.0825	0.0176	21%
CPLE6.SA	0.0017	0.0020	0.1145	0.0124	0.0452	67%
VALE3.SA	0.0022	0.0018	1.9257	0.0290	0.0421	72%
PETR4.SA	0.0031	0.0042	1.7737	0.0473	0.0646	52%
ITSA4.SA	0.0015	0.0012	0.1368	0.0142	0.0343	68%

Fonte: elaborado pelo autor.

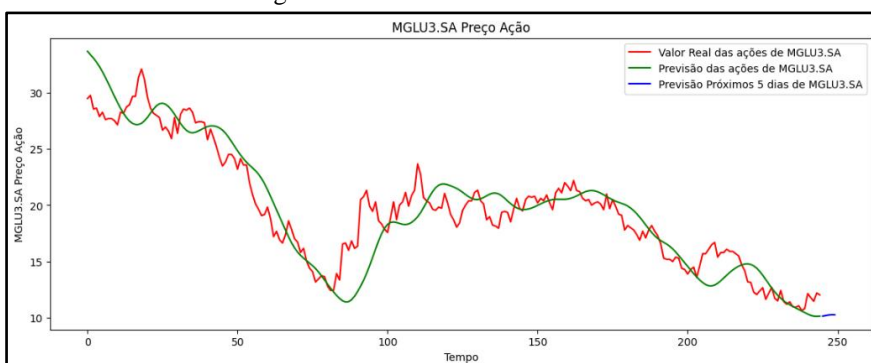
As Figura 6 a Figura 15 mostram a comparação entre os valores reais e os valores preditos por cada modelo. Nas Figura 6 e Figura 7, observamos as previsões para MGLU3, onde ambos os modelos apresentam dificuldade em capturar as rápidas variações do mercado, resultando em previsões menos precisas. As Figura 8 e Figura 9 apresentam as previsões para CPLE6, onde o modelo LSTM mostra uma maior precisão em relação ao GRU, evidenciado pela menor dispersão dos valores preditos em comparação aos reais. Nas Figura 10 e Figura 11 referentes à VALE3, o modelo LSTM demonstra uma maior capacidade de acompanhar as tendências de preços, refletindo uma precisão superior. As Figura 12 e Figura 13 exibem as previsões para PETR4, destacando a eficiência do modelo GRU em determinadas ações. Por fim, as Figura 14 e Figura 15 ilustram as previsões para ITSA4, onde ambos os modelos conseguem capturar bem as tendências, porém o LSTM se sobressai ligeiramente em termos de acurácia.

Figura 6 – Previsão MGLU3 GRU



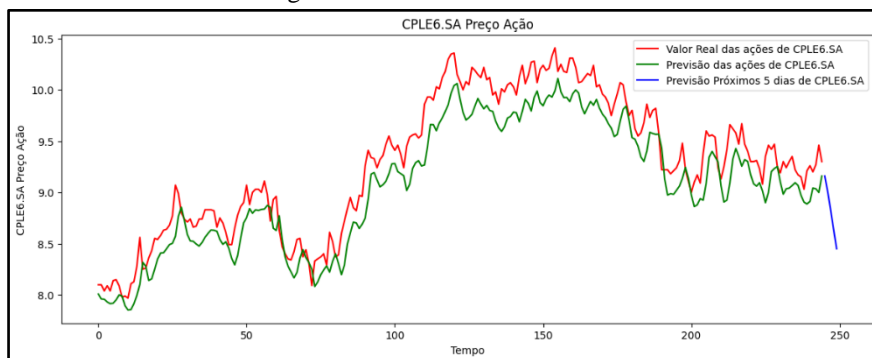
Fonte: elaborado pelo autor.

Figura 7 – Previsão MGLU3 LSTM



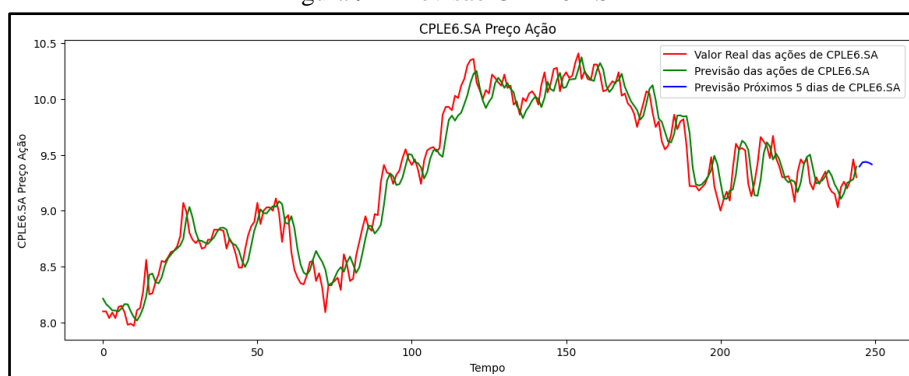
Fonte: elaborado pelo autor.

Figura 8 – Previsão CPLE6 GRU



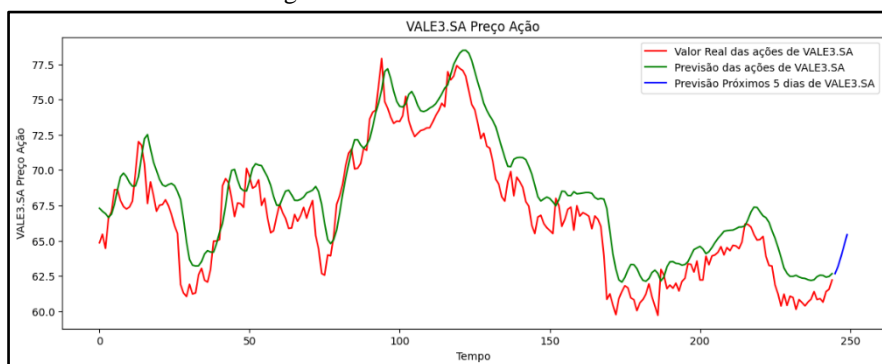
Fonte: elaborado pelo autor.

Figura 9 – Previsão CPLE6 LSTM



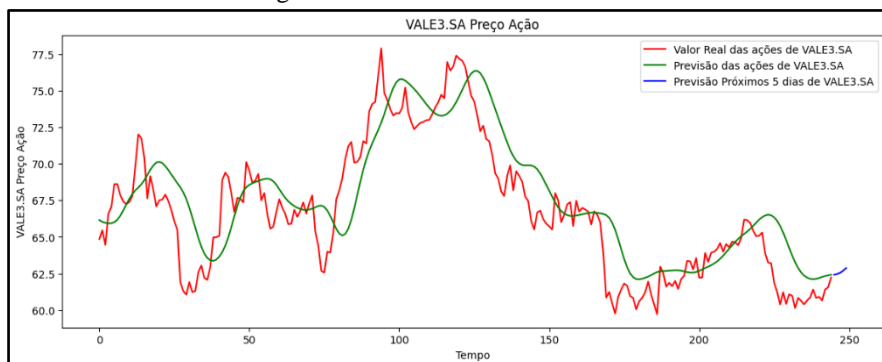
Fonte: elaborado pelo autor.

Figura 10 – Previsão VALE3 GRU



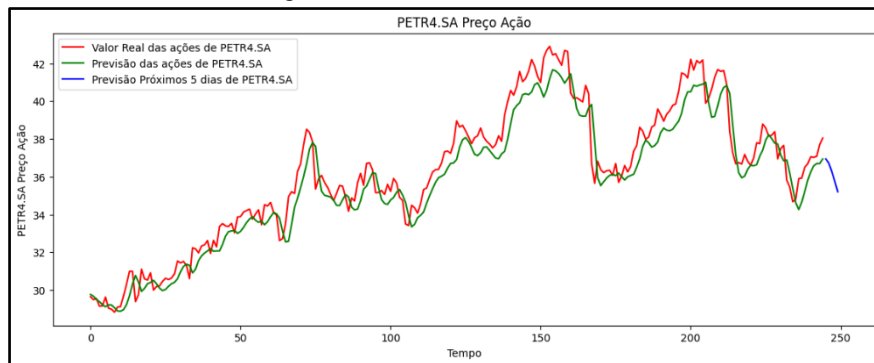
Fonte: elaborado pelo autor.

Figura 11 – Previsão VALE3 LSTM



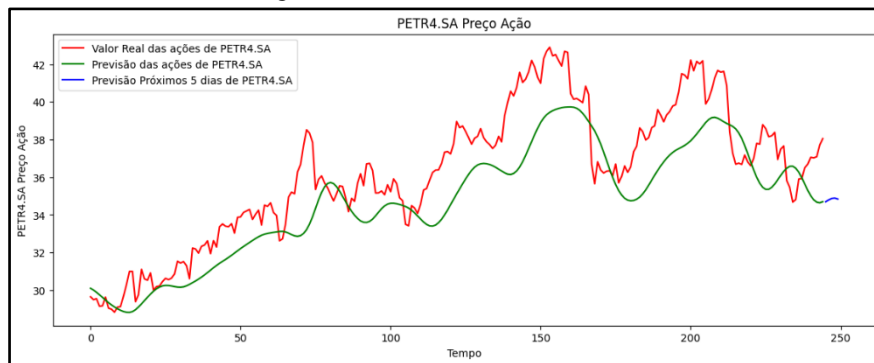
Fonte: elaborado pelo autor.

Figura 12 – Previsão PETR4 GRU



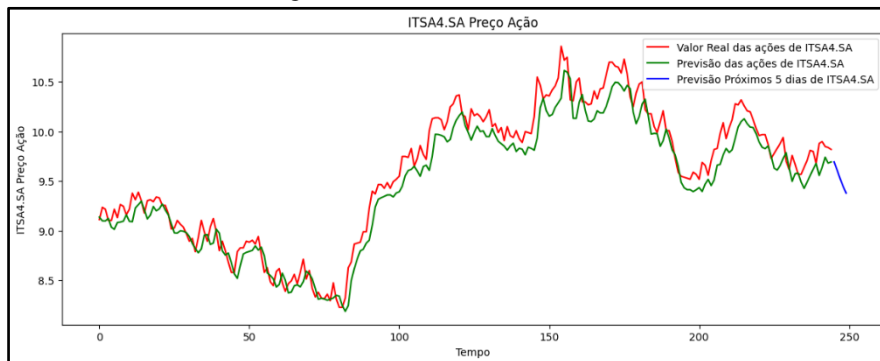
Fonte: elaborado pelo autor.

Figura 13 – Previsão PETR4 LSTM



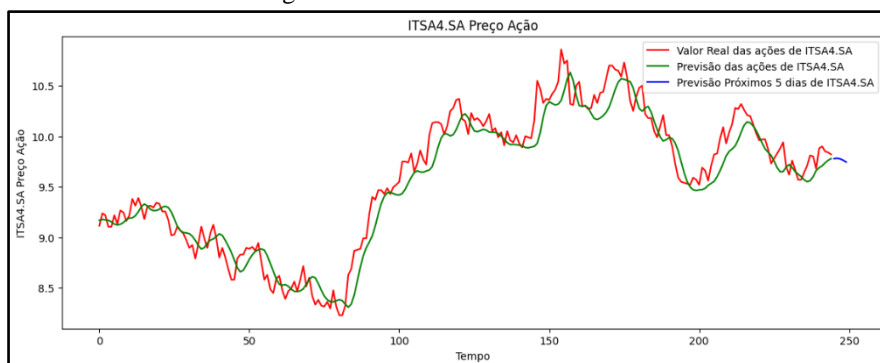
Fonte: elaborado pelo autor.

Figura 14 – Previsão ITSA4 GRU



Fonte: elaborado pelo autor.

Figura 15 – Previsão ITSA4 LSTM



Fonte: elaborado pelo autor.

5 CONCLUSÕES

Através deste estudo, demonstrou-se que tanto os algoritmos LSTM quanto GRU apresentam capacidades robustas para a previsão de valores de ações na bolsa de valores brasileira. Os experimentos realizados mostraram que ambos os modelos podem ser eficazes, sendo possível avaliar o desempenho desses algoritmos em termos de acurácia e outras métricas estatísticas, como MSE, MAE, MAPE e RMSE.

Os resultados indicaram que ambos os algoritmos possuem potencial significativo na previsão de valores de ações, embora apresentem variações em suas performances dependendo das características específicas dos dados de entrada e das empresas analisadas. De maneira geral, o modelo LSTM mostrou-se ligeiramente superior em termos de acurácia, especialmente na previsão dos preços da VALE3, enquanto o modelo GRU demonstrou ser mais eficiente em termos de computação e tempo de treinamento, com destaque para a PETR4.

Com esses resultados, este trabalho demonstra a viabilidade do uso de redes neurais recorrentes para previsões financeiras, oferecendo uma alternativa às abordagens tradicionais de análise técnica. Tanto LSTM quanto GRU podem auxiliar *traders* e investidores a tomarem decisões, também ajustado a comunidade academia com estudos na área financeira.

Durante a execução do projeto, utilizou-se a linguagem de programação Python, com diversas bibliotecas específicas. A coleta dos dados foi realizada com a biblioteca Yfinance, enquanto Numpy, Pandas e Scikit-learn foram empregadas no pré-processamento dos dados. O desenvolvimento e treinamento dos modelos GRU e LSTM foram conduzidos com a biblioteca Keras. Todo o processo de desenvolvimento foi realizado na plataforma Google Colab.

No entanto, algumas limitações foram identificadas, como a sensibilidade dos modelos a eventos inesperados e à volatilidade do mercado, que podem afetar a precisão das previsões. Futuros trabalhos podem explorar a integração de outros fatores macroeconômicos e variáveis de mercado. Por exemplo, pode-se utilizar uma análise fundamentalista que leva em consideração dados da empresa, bem como a sensibilidade do mercado a notícias relevantes para a empresa. Além disso, a extensão do período de análise e a inclusão de mais empresas podem proporcionar uma avaliação mais abrangente e robusta dos algoritmos utilizados.

REFERÊNCIAS

- ALMEIDA, Guilherme. Bolsa de Valores: entenda tudo sobre esse mercado. [2022]. Disponível em: <https://www.certifiquei.com.br/bolsa-valores/>. Acesso em: 20 abr. 2024.
- ArunKumar, K. E., et al. Forecasting of COVID-19 using deep layer recurrent neural networks (RNNs) with gated recurrent units (GRUs) and long short-term memory (LSTM) cells.[2021] Disponível em: <https://www.sciencedirect.com/science/article/pii/S0960077921002149> Acesso em: 29 jun. 2024.
- CAPITALRESEARCH. Bolsa de Valores: o que é, qual sua importância e 5 razões para investir. [2019]. Disponível em: <https://capitalresearch.com.br/blog/bolsa-de-valores/>. Acesso em: 21 jun. 2024.
- CHO, Kyunghyun; MERRIENBOER, Bart van; GULCEHRE, Caglar; BOUGARES, Fethi; SCHWENK, Holger; BENGIO, Yoshua. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics, 2014. p. 1724-1734. Disponível em: <https://www.aclweb.org/anthology/D14-1179.pdf>. Acesso em: 15 mar. 2023.
- Deep Learning Book. Capítulo 52 – Arquitetura de Redes Neurais Gated Recurrent Unit (GRU) Disponível em: <https://www.deeplearningbook.com.br/arquitetura-de-redes-neurais-gated-recurrent-unit-gru/>. Acesso em: 18 abr. 2024.
- FALCÃO, João Vitor Regis, et al. REDES NEURAIAS DEEP LEARNING COM TENSORFLOW [2019] Disponível em: <https://revistas.unifenas.br/index.php/RE3C/article/view/232>. Acesso em: 28 abr. 2024.
- FERNANDES, Marcos da Silva; HAMBERGER, Paula Andréa do Valle; VALLE, Ana Cláudia Marques do. Análise técnica e eficiência dos mercados financeiros: uma avaliação do poder de previsão dos padrões de candlestick. Revista Evidenciação Contábil & Finanças, v. 3, n. 3, p. 35-54, 2015.
- FREIRE, Ananda L.; MENEZES JR, José Maria P. de; BARRETO, Guilherme A. Redes Neurais Recorrentes para predição recursiva de séries temporais caóticas: um estudo comparativo. 2009. Disponível em: https://sbic.org.br/wp-content/uploads/sites/4/2016/10/048_CBR. Acesso em: 18 abr. 2024.
- HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long short-term memory. Neural Computation, v. 9, n. 8, p. 1735-1780, 1997. [1997]. Disponível em: <https://www.bioinf.jku.at/publications/older/2604.pdf>. Acesso em: 18 abr. 2024.
- GOULÃO, Gonçalo José Francisco. Análise técnica no PSI20 MACD, RSI e Estocástico.[2017]. Disponível em: <https://www.iscte-iul.pt/thesis/5885>. Acesso em: 18 abr. 2024.

MARTINS, Carlos. Os supersinais da análise técnica. Elsevier Science, 2012.

MATSUMOTO, Daniel K.F. Estudo em Séries Temporais Financeiras utilizando Redes Neurais Recorrentes. [2019]. Trabalho de Conclusão de Curso (Programa de Pós-graduação em modelagem do Conhecimento) - Instituto de Computação, Universidade Federal de Alagoas, Alagoas

MATSUMOTO, Fernando. Redes Neurais | LSTM Como lidar com perda de memória com Machine Learning. [2019]. Disponível em: <https://medium.com/turing-talks/turing-talks-27-modelos-de-predi%C3%A7%C3%A3o-lstm-df85d87ad210>. Acesso em: 20 abr. 2024.

MITTAL, Aditi. Compreendendo RNN e LSTM [2019]. Disponível em: <https://aditi-mittal.medium.com/entendendo-rnn-e-lstm-f7cdf>. Acesso em 22 mar. 2024.

MODALMAIS. Análise gráfica: o que é e como fazer. [2013]. Disponível em: <https://www.modalmais.com.br/blog/analise-grafica/>. Acesso em: 20 jun. 2024

NADAI, Bruno. Bandas de Bollinger: como usar e interpretar? [2024]. Disponível em: <https://www.infomoney.com.br/guias/bandas-bollinger/>. Acesso em: 20 abr. 2024.

NELSON, David Michael Quirino. Uso de redes neurais recorrentes para previsão de séries temporais financeiras. [2017]. Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais

PEREIRA, Matheus de Mattos, Aprendizado profundo: redes LSTM [2017]. Trabalho de Conclusão de Curso aprovado como requisito para obtenção do título de Bacharel em Sistemas de Informação na Universidade Federal da Grande Dourados.

PINTO, Leonardo. O que é análise técnica: confira o guia completo. [2020]. Disponível em: <https://conteudos.xpi.com.br/aprenda-a-investir/relatorios/o-que-e-analise-tecnica/>. Acesso em: 22 abr. 2024.

PINHEIRO, Layonel. Análise Técnica: o que é e como usar no mercado de ações [2020]. Disponível em: <https://blog.nelogica.com.br/analise-tecnica/>. Acesso em: 12 mar. 2024

REIS, Tiago. Análise de ações: quais são os principais métodos de análise? [2020]. Disponível em: <https://www.suno.com.br/artigos/analise-de-acoes/>. Acesso em: 12 jun. 2024.

RIBEIRO, Marcos Vinicius de Oliveira; CORREA, Matheus Morgado. Previsão de Preço de Ações Baseada em Redes Neurais Recorrentes LSTM e GRU. [2021]. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação e Bacharelado em Sistema da Informação) - Instituto de Computação, Universidade Federal Fluminense, Niterói

ROSSI, Rafael Bertoldi. Protótipo para previsão de ações utilizando redes neurais artificiais. [2019]. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

RUMELHART, David E.; HINTON, Geoffrey E.; WILLIAMS, Ronald J. Learning representations by back-propagating errors. [1986]. Disponível em: <https://cepdnackl.github.io/co542-neural-networks-reading-group-e15/slides/03>. Acesso em: 05 jun. 2024.

SANTODIGITAL. O que é Google Colab? Conheça os benefícios para colaboração em projetos e análise de dados. [2023]. Disponível em: <https://santodigital.com.br/google-colab/>. Acesso em: 30 jun. 2024.

SILVA, Jéssica Aparecida. A importância de padronizar seus dados em Machine Learning. [2023]. Disponível em: <https://medium.com/@jeapsilva/quando-utilizar-standardscaler-minmaxscaler-e-robustscaler-em-machine-learning-788de8b157ca>. Acesso em: 20 abr. 2024.

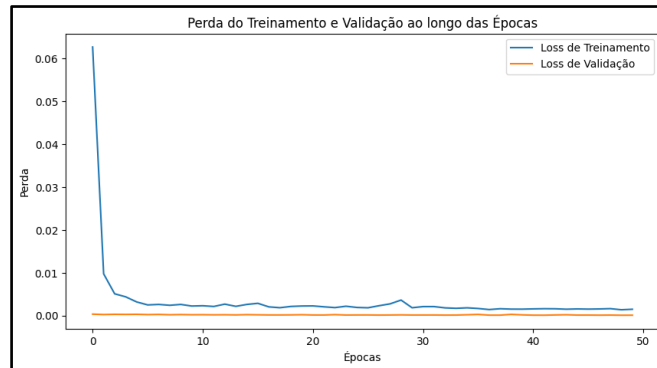
TORRES, Sérgio. Previsão do preço de ações brasileiras utilizando redes neurais artificiais. [2021]. Trabalho de Conclusão de Curso (Pós-graduação em Administração de Empresas), Universidades Presbiteriana Mackenzie, São Paulo

XP Education. O machine learning pode ser utilizado no mercado financeiro? Vem que a gente explica! [S.l.], [2023]. Disponível em: <https://blog.xpeducacao.com.br/machine-learning-no-mercado-financeiro/>. Acesso em: 22 abr. 2024.

6 APÊNDICE A – GRÁFICOS DE PERDA

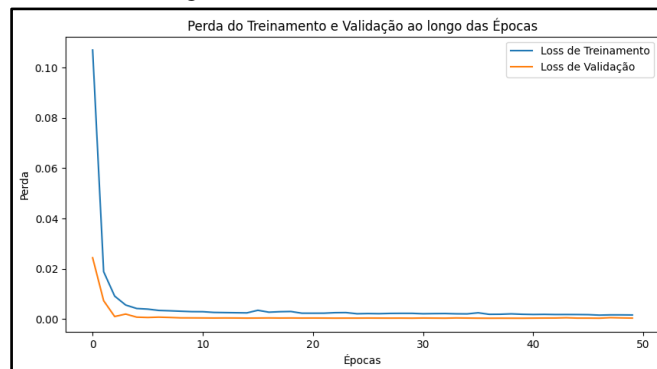
Abaixo os gráficos mostrando a o resultado do MSE durante a fase de treinamento e de validação das seguintes empresas MGLU3, VALE3, PETR4 e ITSA4.

Figura 16 - MSE MGLU3 GRU



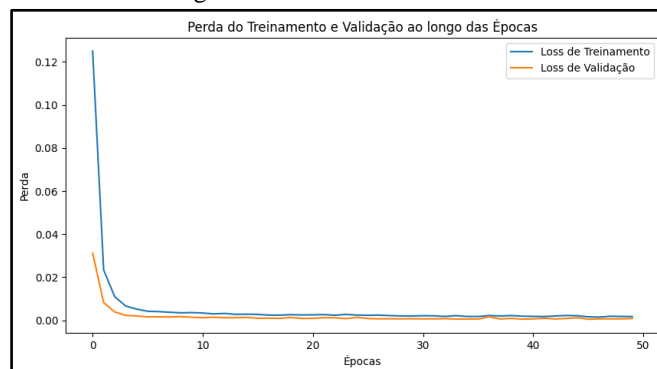
Fonte: elaborado pelo autor.

Figura 17- MSE MGLU3 LSTM



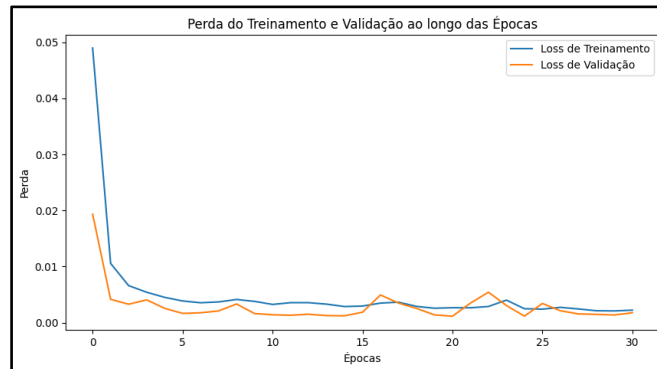
Fonte: elaborado pelo autor.

Figura 18 - MSE VALE3 GRU



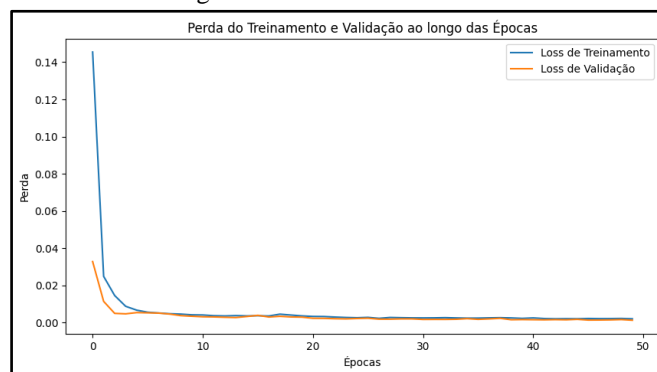
Fonte: elaborado pelo autor.

Figura 19 MSE VALE3 LSTM



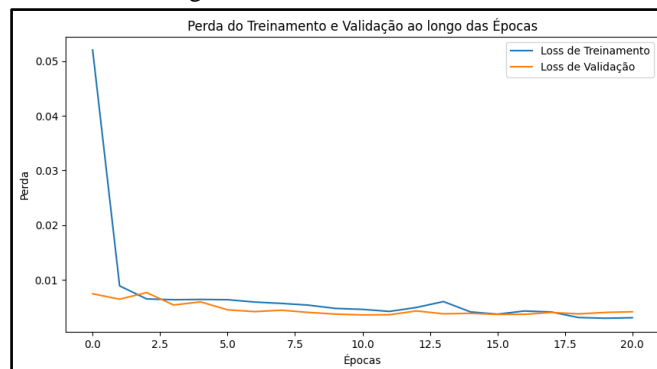
Fonte: elaborado pelo autor.

Figura 20 - MSE PETR4 GRU



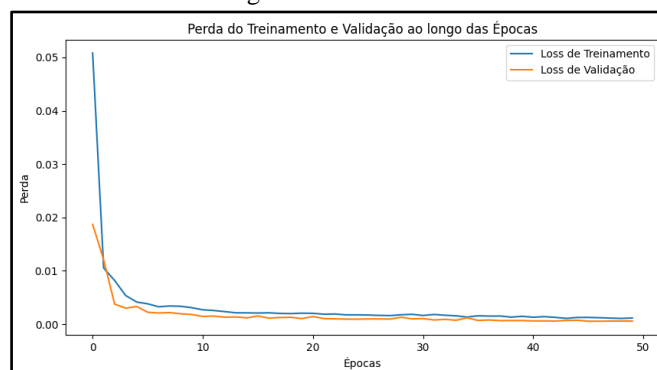
Fonte: elaborado pelo autor.

Figura 21 - MSE PETR4 LSTM



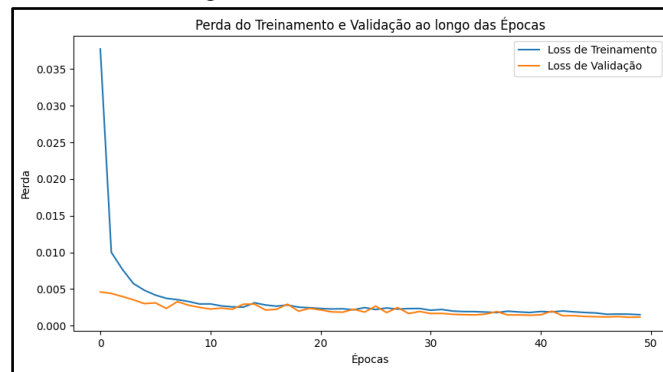
Fonte: elaborado pelo autor.

Figura 22 - MSE ITSA4 GRU



Fonte: elaborado pelo autor.

Figura 23 MSE ITSA4 LSTM



Fonte: elaborado pelo autor.