



UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Tecnologia

Cléofas Peres Santos

**UTILIZAÇÃO DE REDES NEURAIS ARTIFICIAIS PARA A
PREDIÇÃO DO PREÇO DO BITCOIN ATRAVÉS DE
SÉRIES TEMPORAIS**

Limeira
2021

Cléofas Peres Santos

**UTILIZAÇÃO DE REDES NEURAIS ARTIFICIAIS PARA A PREDIÇÃO
DO PREÇO DO BITCOIN ATRAVÉS DE SÉRIES TEMPORAIS**

Monografia apresentada à Faculdade de Tecnologia da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Bacharel em Sistemas de Informação, na área de Sistemas de Informação e Comunicação.

Orientador: Prof. Dr. João Roberto Bertini Junior

Este exemplar corresponde à versão final da Monografia defendida por Cléofas Peres Santos e orientada pelo Prof. Dr. João Roberto Bertini Junior.

Limeira
2021

FOLHA DE APROVAÇÃO

Abaixo se apresentam os membros da comissão julgadora da monografia em defesa do Título de Bacharel em Sistemas de Informação na área de concentração Sistemas de Informação e Comunicação, a que se submeteu o aluno Cléofas Peres Santos, em 04 de janeiro de 2021 na Faculdade de Tecnologia – FT/UNICAMP, em Limeira/SP.

Prof. Dr. João Roberto Bertini Junior
Presidente da Comissão Julgadora

Prof. Dr. Plínio Roberto Souza Vilela
FT/UNICAMP

Prof. Dr. Guilherme Palermo Coelho
FT/UNICAMP

Ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria de Graduação da FT.

Agradecimentos

Agradeço a Deus por minha vida e por conseguir chegar onde cheguei. Agradeço ao meu orientador João Roberto Bertini Junior por aceitar conduzir o meu trabalho de pesquisa. Agradeço a todos os meus professores do curso de Sistemas de Informação da Universidade Estadual de Campinas pela excelência da qualidade técnica de cada um. Agradeço a todos os meus amigos do curso de graduação que compartilharam dos inúmeros desafios que enfrentamos, sempre com o espírito colaborativo.

Resumo

O Bitcoin é atualmente a moeda digital mais popular, mais negociada (alta liquidez) e a maior do ramo, com relação à capitalização. Esses fatores a torna uma opção atrativa para os investidores. Surge então a necessidade de criar modelos de predição para a mesma. Uma maneira eficiente de criar tais modelos é o emprego das redes neurais artificiais. Desta forma, este trabalho tem por objetivo apresentar os resultados alcançados com a aplicação de uma rede neural recorrente na predição da série temporal do Bitcoin. Para esta pesquisa, foram desenvolvidos dois algoritmos, o primeiro tendo apenas um previsor (valor de fechamento diário) e o segundo com diversos previsores (valor de abertura, valor máximo atingido, valor mínimo atingido, valor de fechamento e fechamento ajustado). O algoritmo com melhor desempenho foi o segundo, apresentando uma taxa de erro de apenas 2,01%.

Abstract

Bitcoin is currently the most popular, most traded (high liquidity) digital currency and the largest in the industry, with respect to capitalization. These factors make it an attractive option for investors. Then the need arises to create prediction models for it. An efficient way to create such models is to use artificial neural networks. Thus, this work aims to present the results achieved with the application of a recurrent neural network in the prediction of the Bitcoin time series. For this research, two algorithms were developed, the first having only one predictor (daily closing value) and the second with several predictors (opening value, maximum value reached, minimum value reached, closing value and adjusted closing. the best performance was the second, presenting an error rate of only 2.01 %.

Lista de Figuras

3.1	Representação de um neurônio biológico.	19
3.2	Representação de um neurônio artificial	19
3.3	Representação gráfica de uma RNA	21
3.4	Diferença entre RNN e Feedforward	22
3.5	Representação de uma RNN	23
3.6	Estrutura dos gates	23
3.7	Esquema gráfico LSTM	24
4.1	Representação do modelo de arquitetura proposta.	32
5.1	Leitura do dataframe	34
5.2	Seleção da coluna Close	34
5.3	Seleção das colunas Open, High, Low, Close e Adj Close	34
5.4	Normalização dos dados algoritmo 1	35
5.5	Normalização dos dados Normalização dos dados algoritmo 2	35
5.6	Modelagem do dataframe para o algoritmo 1	35
5.7	Modelagem do dataframe para o algoritmo 2	36
5.8	Criação da base de treinamento	36
5.9	Estrutura da rede neural	37
5.10	Curva de previsão algoritmo 1	38
5.11	Decaimento do MSE no treinamento da rede	38
5.12	Curva de previsão algoritmo 2	39
5.13	Decaimento do MSE no treinamento da rede	40

Lista de Tabelas

5.1	Resultados da predição realizada pelo algoritmo 1.	39
5.2	Resultados da predição realizada pelo algoritmo 2.	40

Lista de símbolos e siglas

RNA	Rede Neural Artificial
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
API	Application Programming Interface
PDF	Portable Document Format
ARIMA	Autoregressive Integrated Moving Average
CSV	Comma-separated-values

Sumário

1	Introdução	12
1.1	Justificativa	13
1.2	Objetivos	14
1.2.1	Objetivo geral	14
1.2.2	Objetivos específicos	14
1.3	Estrutura do documento	15
2	Levantamento bibliográfico	16
3	Fundamentação teórica	18
3.1	Fundamento de redes neurais artificiais	18
3.1.1	Neurônios biológicos	18
3.1.2	Neurônio artificial	19
3.1.3	Redes neurais artificiais	20
3.1.4	Deep Learning	21
3.1.5	Redes Neurais Recorrentes	22
3.2	Blockchain	24
3.3	Criptomoedas	25
3.3.1	O Bitcoin	26
3.4	Séries temporais	27
4	Ferramentas e Métodos	28
4.1	Seleção da criptomoeda	28
4.2	Linguagem de programação Python	28
4.3	Biblioteca Pandas	29
4.4	Biblioteca Matplotlib	29
4.5	Biblioteca Sklearn	29
4.6	Especificação do modelo de rede neural artificial	30
4.6.1	Definição da arquitetura	30
4.7	Configuração de Hardware	32
5	Desenvolvimento	33
5.1	Coleta de dados	33
5.2	Pré-processamento	34
5.3	Desenvolvimento da rede neural artificial	36
5.4	Análise dos Resultados	37
5.4.1	Aplicação do primeiro algoritmo (1 entrada)	37
5.4.2	Aplicação do segundo algoritmo (múltiplos previsores)	39

6 Conclusões	41
Referências bibliográficas	43

Capítulo 1

Introdução

O mundo está em constante evolução, as relações comerciais e as formas de pagamentos não são diferentes. O que começou com troca direta de mercadorias (escambo), evoluiu para trocas mediadas, onde produtos de amplo interesse (sal, cereais, gado e por fim metais preciosos) eram utilizados como moeda de troca.

Com o passar do tempo, esses produtos foram substituídos por notas (moeda e papel), dispensando o esforço de carregar grandes quantidades de mercadorias para realizar trocas. Inicialmente a emissão de papel era totalmente baseada na quantidade de matérias preciosas disponíveis, e sua conversão era total. Com o desenvolvimento das economias, as moedas deixaram de ser totalmente lastreadas, tornando-se moeda fiduciária ou seja, sem valor intrínseco, seu valor advém da confiança entre os agentes e em quem emitiu o título. No entanto, esse sistema ruiu, seja em consequência da demasiada emissões de moeda, seja pela baixa correspondência na disponibilidade de ouro, o que levou o Estado a controlar a emissão de moeda, surgindo assim as moedas de curso forçado, com inexistência de lastro metálico, inconvertibilidade absoluta e com monopólio estatal.

O avanço tecnológico e o desenvolvimento da internet propiciaram novas alternativas ao uso das moedas convencionais. Os agentes econômicos começaram a perder o contato físico com o dinheiro e a fazerem as transações via canais eletrônicos, via cartão de débito e crédito, transferências bancárias eletrônicas e modalidades como o PagSeguro¹, Paypal², ou PicPay³.

¹PagSeguro sistema de pagamento eletrônico para compra de mercadorias para pessoas físicas ou jurídicas efetuarem compras de forma segura via internet.

²Paypal é um sistema que permite a transferência de dinheiro entre indivíduos usando um endereço de e-mail, assim, evitando métodos tradicionais como cheques e boleto bancário.

³PicPay é um aplicativo disponível Android e iPhone que funciona como uma carteira digital. Todas as transferências são feitas de forma digital, sem contato físico.

Esses meios de transações comerciais possibilitam a realização de compras em qualquer lugar do mundo e representam a desmaterialização da moeda, com o contato físico sendo cada vez menor.

Mais recentemente surgem novas opções monetárias como as moedas digitais, que ao contrário das moedas tradicionais, são descentralizadas, não são controladas por nenhum órgão estatal e são mantidas pelos próprios usuários. Em meio a diversas opções, destaca-se a criptomoeda Bitcoin, com maior aceitação, popularidade e liquidez (alto número de transações), moeda essa que será objeto de pesquisa desse estudo.

O Bitcoin foi apresentado ao mundo no ano de 2008, quando Satoshi Nakamoto⁴ publica um paper⁵ e o lança na internet explicando a criação e o funcionamento da referida criptomoeda. Desde então, o Bitcoin vem ganhando força, e se faz cada vez mais presente em carteiras de investimentos, seja para Hedge⁶, ou para lucrar com sua valorização.

Tal popularidade e potencial de valorização são responsáveis pelo desenvolvimento de inúmeros estudos nesta área. Dentre as pesquisas referentes à predição de cotações futuras que utilizam séries temporais, as redes neurais artificiais têm se destacado. A capacidade destas técnicas em trabalhar com uma quantidade significativa de variáveis simultâneas, além da composição de sua estrutura maciçamente paralela e distribuída, evidenciam sua alta escala de poder computacional, concedendo as habilidades de aprendizado e generalização de funções. Estas duas capacidades de processamento de informação, tornam possíveis para as RNAs resolver problemas de grande escala que, usando o processamento digital convencional, são consideradas computacionalmente inviáveis e intratáveis. (HAYKIN, 2000)

Neste sentido, tendo em vista a importância das redes neurais aplicadas na predição de eventos futuros, o presente trabalho apresenta um estudo da aplicação destas técnicas na previsão de preços futuros da mais relevante criptomoeda, o Bitcoin.

1.1 Justificativa

São diversas as questões que levam à necessidade de previsão do preço Bitcoin. O principal motivo é garantir maior lucratividade possível com a execução de bons investimentos,

⁴Satoshi Nakamoto é o pseudônimo utilizado pelo(s) criador(es) do sistema Bitcoin, cujo nome e nacionalidade reais ainda são desconhecidos.

⁵Disponível em: bitcoin.org/bitcoin.pdf

⁶Hedge é uma estratégia de proteção para os riscos de um investimento, que neutraliza a posição comprada ou vendida para que seu preço não varie.

fazendo com que os investidores obtenham sempre uma vantagem sobre os demais que atuam negociando outros ativos.

Um outro ponto que deve ser destacado e que influenciou na decisão de realizar este trabalho aplicado ao mercado de criptomoedas é a distribuição do modelo de dados em que o Bitcoin opera, através de oscilações diárias, formando assim, séries temporais. Tendo isso em vista, este cenário do mercado de criptomoedas proporciona diretrizes para o aperfeiçoamento de arquiteturas de RNAs que utilizam um modelo de dados baseado em séries temporais.

Por fim, o vasto crescimento de aplicações que utilizam o auxílio de tecnologias baseadas em técnicas de inteligência artificial também foi um fator determinante para definir a aplicabilidade deste trabalho.

1.2 Objetivos

Com base no tema definido, estabelecem-se os objetivos a serem atingidos ao final deste estudo. Desta forma temos, respectivamente, o objetivo principal e os objetivos específicos, apresentados em ordem de desenvolvimento.

1.2.1 Objetivo geral

Assume-se, como objetivo principal deste estudo a aplicação de um modelo de rede neural artificial para realizar previsões nos valores de fechamento da criptomoeda Bitcoin.

1.2.2 Objetivos específicos

Com o objetivo geral determinado, tornaram-se necessários as definições de alguns objetivos específicos, sendo:

- Especificar uma arquitetura de RNA;
- Treinar o modelo de RNA especificado;
- Realizar testes com o modelo treinado;
- Analisar os resultados obtidos pela implementação, comparando-os com os seus resultados reais e avaliando sua capacidade de acerto.

1.3 Estrutura do documento

O documento foi dividido em seis capítulos: introdução, levantamento bibliográfico, fundamentação teórica, ferramentas e métodos, desenvolvimento e conclusões.

O atual capítulo contextua e inicia o tema a ser tratado, bem como estabelece os objetivos a serem atingidos.

No segundo capítulo, são apontados os trabalhos que foram utilizados como referência e motivação. A fundamentação teórica, apresentada no terceiro capítulo, descreve o embasamento referente aos conceitos das técnicas que serão utilizadas para a realização deste estudo.

O quarto capítulo é destinado ao detalhamento dos métodos, técnicas e procedimentos que têm por objetivo gerar conhecimento para viabilizar um melhor ambiente de desenvolvimento para o trabalho.

O quinto capítulo detalha a implementação dos métodos, que segue as especificações das técnicas levantadas no capítulo anterior.

Enfim, no sexto capítulo são tratadas as conclusões obtidas pela técnica de RNA desenvolvida.

Capítulo 2

Levantamento bibliográfico

Esta seção visa apresentar brevemente os estudos anteriores sobre a previsão de cotações futuras do Bitcoin usando-se de técnicas de inteligência artificial. São inúmeras as técnicas de inteligência artificial que vem sendo desenvolvidas e aperfeiçoadas ao longo do tempo, com o objetivo de possibilitar predições de boa acurácia, tal como o estudo realizado por (ALESSANDRETTI et al., 2018) que propôs dois modelos de previsão baseados em redes neurais recorrentes (RNNs) e memória de curto longo prazo (LSTM), e os comparou com um modelo de média móvel integrada autoregressiva (ARIMA), que é amplamente utilizada modelo de previsão.

No trabalho dos autores (MCNALLY; ROCHE; CATON, 2018), os modelos RNN e LSTM mostraram-se melhores do que o modelo ARIMA. Além das informações de preço, Saad e Mohaisen (SAAD et al., 2019) analisaram as informações de blockchain, como o número de carteiras Bitcoin e endereços únicos, dificuldade de mineração de bloco, taxa de hash, etc., e usaram aqueles recursos que são altamente correlacionados com o preço do Bitcoin para construir modelos de previsão.

Métodos híbridos vêm ganhando cada vez mais espaço entre as pesquisa mais recentes. Na pesquisa produzida por (KHANDELWAL; ADHIKARI; VERMA, 2015) foi utilizado uma abordagem que mescla características do modelo ARIMA com o de uma RNA. Os pesquisadores testaram os métodos ARIMA e RNA para isolar componentes lineares e não-lineares, respectivamente, de séries temporais, com o objetivo de encontrar um melhor padrão para tratar a oscilação dos dados obtidos.

É importante salientar que os trabalhos citados neste capítulo, por alguma peculiaridade ou detalhe técnico serviram como motivação para a idealização deste estudo, não podendo

então deixar de citá-los. Além disso, provaram ser referência dentro de suas respectivas problemáticas, trazendo análises e conclusões significativas, que são essenciais para o aperfeiçoamento acadêmico e científico.

Capítulo 3

Fundamentação teórica

Nas seções posteriores, serão abordados os fundamentos que suportam o tema proposto, portanto, seguem os conceitos chave para compreensão do processo de funcionamento das redes neurais atificiais e a aplicação do modelo na predição do preço do Bitcoin.

3.1 Fundamento de redes neurais artificiais

As redes neurais artificiais (RNA) são modelos matemáticos/computacionais inspirados no comportamento observado em estruturas de organismos inteligentes. São formadas por um conjunto de unidades chamadas neurônios artificiais, conectados entre si. (PAULA et al., 2002). As RNAs buscam simular métodos de aprendizado do cérebro humano, por meio do uso de neurônios artificiais. Uma RNA é baseada nos neurônios biológicos e no sistema nervoso, portanto, para o entendimento da mesma, faz-se necessário a compreensão do funcionamento dos neuronios biológicos e artificiais, assunto abordado a seguir.

3.1.1 Neurônios biológicos

O neurônio é a unidade básica que constitui a estrutura cerebral e do sistema nervoso. Um neurônio é composto por três partes principais: dendritos, corpo celular e axônio. Dendritos são as estruturas ramificadas que recebem o sinal dos neurônios circundantes, onde cada ramificação está conectada a um outro neurônio. Axônio é um cilindro fino que transmite o sinal de um neurônio para outros. No final do axônio, o contato com os dendritos é feito através de uma sinapse. A Figura 3.1 representa a estrutura básica de um neurônio biológico.

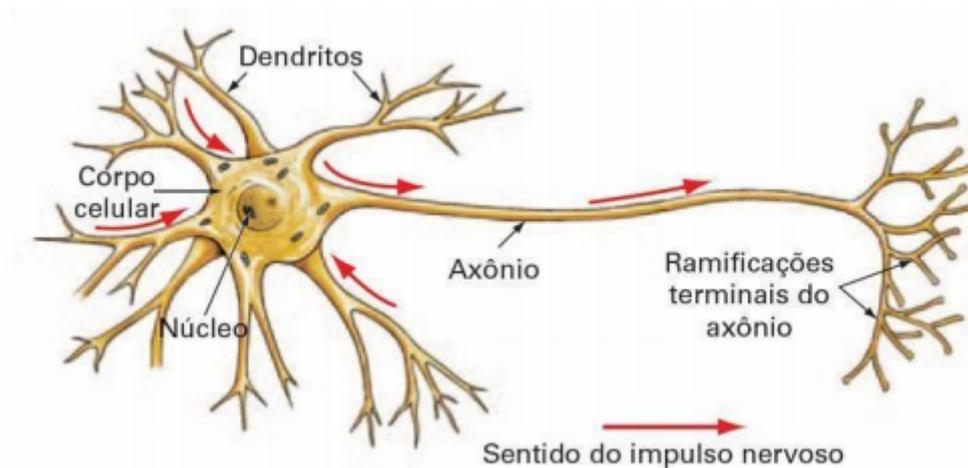


Figura 3.1: Representação de um neurônio biológico.
 Fonte: deeplearningbook.com.br/o-neuronio-biologico-e-matematico

3.1.2 Neurônio artificial

Inspirados no neurônio biológico, em 1943 o psiquiatra e neuroanatomista Warren McCulloch e o matemático Walter Pitts idealizaram o primeiro modelo de neurônio artificial. A estrutura de um neurônio artificial segue os mesmos princípios relacionados aos neurônios biológicos, pretendendo realizar as mesmas funções, utilizando-se de conceitos matemáticos, aritméticos e computacionais. (GOODFELLOW et al., 2016)

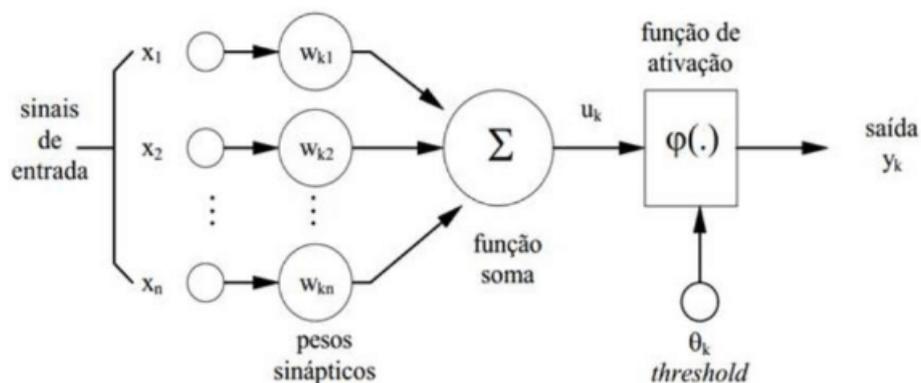


Figura 3.2: Representação de um neurônio artificial.
 Fonte: Adaptado de (HAYKIN, 2007)

A Figura 3.2 esboça a estrutura que compõe um neurônio artificial, demarcando cada item que o compõe. De maneira similar ao neurônio biológico, as entradas (x_1, x_2, \dots, x_m) equivalem aos impulsos elétricos provenientes dos dendritos. Os sinais de entrada são multiplicados por um peso sináptico (representado pela letra W), que revela a relevância

desta entrada para o processamento da saída. Todas as entradas ponderadas pela multiplicação pelos pesos são somadas e então, sujeitadas a uma função de ativação, que define se o neurônio será ativado ou não. A função de ativação representa o limiar de disparo (threshold) de um neurônio biológico. Por fim, temos a saída do neurônio, representada pela letra Y na figura 3.2, que muitas vezes é comparada com uma saída esperada para recalcular os pesos até que a rede esteja devidamente treinada para um determinado modelo esperado.

3.1.3 Redes neurais artificiais

Haykin define redes neurais como sendo ferramentas computacionais, programadas para analisar dados, tal como seu comportamento e relações, visando trabalhar com os mesmos conceitos de um sistema nervoso, simulando o comportamento de um conjunto de neurônios biológicos. Por meio de complexos processos de treinamento e aprendizagem, as RNAs visam realizar o reconhecimento de padrões de dados, para que possa prever resultados, através de classificação ou generalização dos dados em questão. (HAYKIN, 2007).

Tais ferramentas "aprendem" a executar tarefas considerando exemplos, geralmente sem serem programadas com regras específicas. Por exemplo, no reconhecimento de imagem, eles podem aprender a identificar imagens que contenham gatos analisando imagens de exemplo que foram rotuladas manualmente como "gato" ou "sem gato" e usando os resultados para identificar gatos em outras imagens. Esse reconhecimento é possível mesmo não possuindo conhecimento prévio sobre o que caracteriza o fenótipo de um gato, e sim, identificam características do material de aprendizagem que processaram (SILVA; OLIVEIRA; LUDERMIR, 2016).

Devido à sua capacidade de reproduzir e modelar processos, as redes neurais artificiais encontraram aplicações em muitas disciplinas. As áreas de aplicação incluem identificação e controle de sistemas (controle de veículo, previsão de trajetória, controle de processo, gerenciamento de recursos naturais), química quântica, jogos em geral, reconhecimento de padrões (sistemas de radar, identificação de faces, classificação de sinais, reconstrução 3D, reconhecimento de objetos e mais), reconhecimento de sequência (gesto, fala, manuscrito e reconhecimento de texto impresso), diagnóstico médico, finanças (por exemplo, sistemas de negociação automatizados), mineração de dados, visualização, tradução automática, filtragem de redes sociais e filtragem de spam por e-mail. (GOODFELLOW et al., 2016).

O macro fluxo de treinamento de uma rede neural é o seguinte: os neurônios recebem as informações imputadas e propagam o resultado até o fim da rede, após isso, calcula-se o erro da saída processada com o valor desejado, devolvendo o resultado para o início da rede, ou seja, invertendo todo o processo para o ajuste dos pesos. Quanto menor for a taxa de variação do erro, mais próximo do resultado desejado. Abaixo, segue a representação de uma rede neural artificial, onde os nós do grafo simbolizam os neurônios, as arestas significam as conexões e as setas representam a sentido do fluxo de sinal.

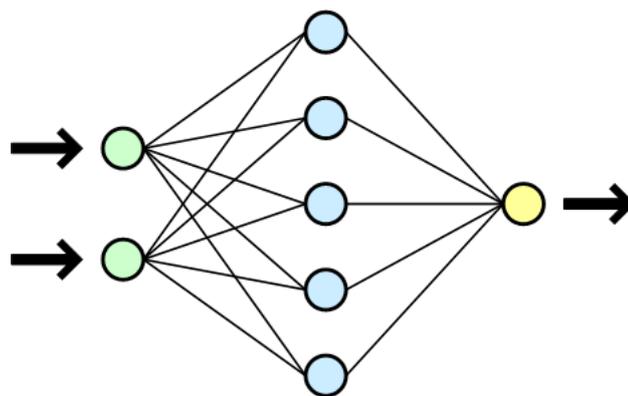


Figura 3.3: Representação gráfica de uma RNA.
Fonte: pt.wikipedia.org/wiki/Rede_neural_artificial

3.1.4 Deep Learning

Como colocam (GOODFELLOW et al., 2016), aprendizagem em profundidade é uma abordagem de inteligência artificial permite a própria rede a aprender com a experiência e entender o mundo a partir de uma hierarquia de conceitos, relacionando cada dado ao conceito mais simples.

Esse é um tema emergente dentro da inteligência artificial (AI) que imita o funcionamento do cérebro humano no processamento de dados e na criação de padrões para uso na tomada de decisões. Esse subconjunto do aprendizado de máquina em inteligência artificial, possui redes capazes de aprender sem supervisão a partir de dados não estruturados ou não rotulados. Também conhecido como aprendizado neural profundo ou rede neural profunda.(AGUIAR, 2010).

3.1.5 Redes Neurais Recorrentes

O homem é capaz de projetar e se antecipar diante de situações sequenciais desde que ele tenha a capacidade de detectar o padrão sequencial de algum evento contínuo. São diversos os exemplos de eventos sequenciais que podemos observar: frames de um vídeo, palavras dentro de uma frase com sentido, o preço de um título na bolsa de valores, a temperatura de uma região em um determinado horário, a taxa de juros ou as vendas realizadas em um comércio. Buscando maneiras eficientes de se realizar previsões sobre dados sequenciais e valendo-se da estrutura de redes neurais artificiais, surgiram as redes neurais recorrentes.

Redes neurais recorrentes (RNN) são uma classe de redes neurais que são naturalmente adequadas para o processamento de dados de séries temporais e outros dados sequenciais. Ao contrário das redes neurais feedforward, RNNs podem usar seu estado interno (memória) para processar sequências de entradas. (ALPAYDIN, 2010)

As RNNs são capazes de armazenar e processar informações no tempo, característica essencial para realizar previsões e garantir uma memória de curto prazo. Esta característica é possível pelo fato de que a informação transmitida de um neurônio de uma camada para o neurônio da camada seguinte também é retransmitida para ele mesmo. Assim, o neurônio possui não apenas informações das entradas atuais, mas também informações das entradas anteriores, como pode-se observar na Figura 3.4, que compara a estrutura da redes recorrente e feedforward.

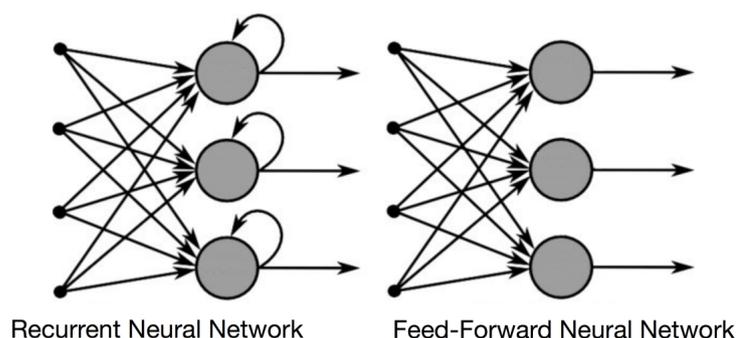


Figura 3.4: Diferença entre RNN e Feedforward.
Fonte: deeplearningbook.com.br/redes-neurais-recorrentes

A Figura 3.5 é a representação de uma camada de RNN no decorrer do tempo. Note a combinação do x com $h(t-1)$ (estado anterior) e a passagem do valor pela função de ativação \tanh , produzindo uma saída $h(t)$ que passará ao estado seguinte.

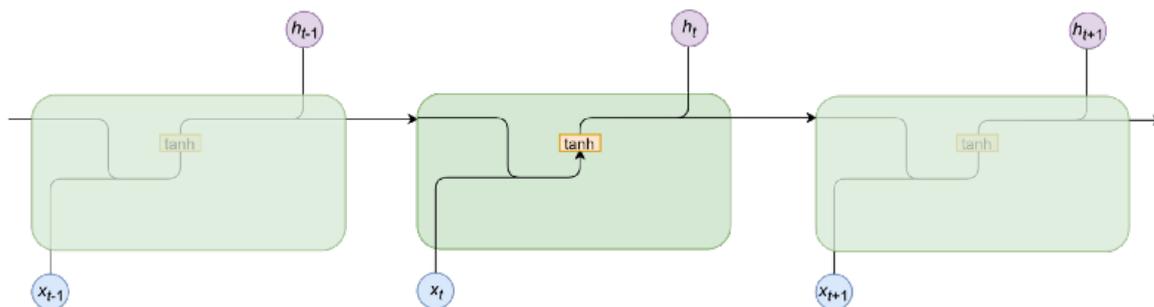


Figura 3.5: Representação de uma RNN.

Fonte: cutt.ly/zjAvSi2

3.1.5.1 Long Short-term Memory - LSTM

Assim como as RNN são uma evolução das redes neurais *Feedforward*, *Long Short-term Memory* é uma versão aprimorada das RNN convencionais, que resolve o problema de dissipação de gradiente (diminuição exponencial do gradiente, evita a mudança dos valores dos pesos, atrapalhando o treinamento da rede) e adiciona a capacidade de guardar informações por longos períodos de tempo. Para isso, são empregadas as portas (ou *gates*) que controlam a passagem de dados entre os estados. O componente básico dessas *gates* é representada pela Figura 3.6:

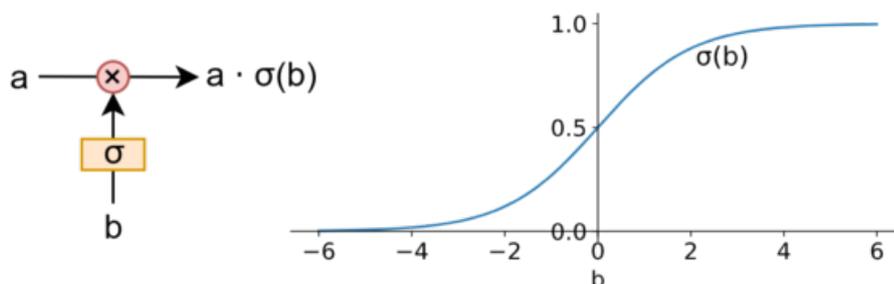


Figura 3.6: Estrutura dos gates.

Fonte: cutt.ly/zjAvSi2

A estrutura é responsável por controlar a porcentagem de cada valor que passará pela porta, esse controle é dado pela aplicação da função sigmóide ao input b e multiplicação do resultado pelo sinal da linha horizontal a , que representa a *cell state*. Na LSTM, tem-se 3 gates, em ordem, são elas:

- *Forget gate*, responsável por decidir quais partes do *cell state* continuam relevantes;
- *Remember gate*, responsável por decidir quais dados da memória de curto prazo devem ser incorporado ao *cell state*;

- *Output gate*, responsável por decidir quais partes do cell state são relevantes no estado atual para produzir o output.

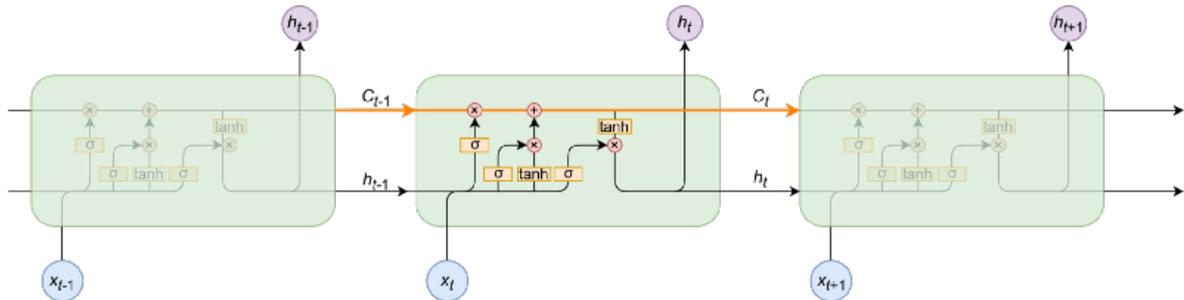


Figura 3.7: Esquema gráfico LSTM.

Fonte: cutt.ly/zjAvSi2

Analisando o esquema acima, observa-se que existem duas linhas principais. A linha em laranja é denominada de célula de memória, pois guarda o *cell state* (memória de longo prazo). A de baixo, correspondente à memória de curto prazo, contém a entrada e o hidden state (a saída no estado anterior). Entre as duas linhas, encontra-se as 3 gates que gerenciam a passagem de dados.

A *forget gate* é responsável por selecionar quais partes do *cell state* são relevantes e quais devem ser inutilizadas. Logo após, seleciona-se novos dados da memória de curto prazo que devem compor no *cell state*, por meio da *input gate*. O último passo é utilizar o *cell state* para calcular a saída h da rede. Para isso, primeiramente, passa-se o estado C por uma função de ativação \tanh . Em seguida, a memória de curto prazo é utilizada para determinar quais partes dessa saída são realmente relevantes nesse instante (*output gate*).

3.2 Blockchain

A recente tecnologia conhecida como Blockchain surgiu junto da criptomoeda Bitcoin, em 2008. Ainda hoje, é comum que associem os dois termos e acabem causando confusão, o blockchain na verdade é a tecnologia que suporta o funcionamento do bitcoin.

Cadeia de bloco, originalmente blockchain, é apenas uma estrutura de dados. Isto é, como os dados são logicamente organizados e armazenados. Os chamados blocos, são ligados usando criptografia, sendo que cada bloco contém um hash criptográfico do bloco anterior, um registro de data e hora e dados de transação (geralmente representados como um hash de raiz de árvore merkle) (HENDRIK WITTE, 2016).

Uma vez registrados, os dados em qualquer bloco não podem ser alterados retroativamente sem alteração de todos os blocos subsequentes, o que requer consenso da maioria da rede. Embora os registros blockchain não sejam inalteráveis, blockchains podem ser considerados seguros pela forma como são modeladas e exemplificam um sistema de computação distribuída com baixa tolerância a falhas. (SCOTT, 2016).

Blockchain foi concebido por Satoshi Nakamoto em 2008 para servir como razão de transação pública da criptomoeda bitcoin. A invenção do blockchain para o bitcoin fez dele a primeira moeda digital a resolver o problema do gasto duplo sem a necessidade de uma autoridade confiável ou servidor central.

As palavras bloco e cadeia foram usadas separadamente no artigo original de Satoshi Nakamoto, mas acabaram sendo popularizadas como uma única palavra, Blockchain. (GUPTA, 2017).

3.3 Criptomoedas

Uma criptomoeda é uma moeda virtual protegida por criptografia, o que torna quase impossível falsificar ou fraudar. As criptomoedas são baseadas na tecnologia blockchain - uma cadeia de blocos de dados distribuída, imposto por uma rede díspar de computadores, como explanado anteriormente. Uma característica definidora das criptomoedas é que elas geralmente não são emitidas por nenhuma autoridade central, tornando-as teoricamente imunes à interferência ou manipulação governamental.

A primeira criptomoeda baseada em blockchain foi o Bitcoin, que ainda continua sendo a mais popular e mais valiosa. Hoje, existem milhares de criptomoedas alternativas com várias funções e especificações. Algumas dessas são baseadas no Bitcoin, enquanto outras são novas moedas que foram construídas a partir do zero (PAAR; PELZL, 2009).

As criptomoedas mantêm a promessa de facilitar a transferência de fundos diretamente entre duas partes, sem a necessidade de terceiros confiáveis, como um banco ou empresa de cartão de crédito. Essas transferências são protegidas pelo uso de chaves públicas e chaves privadas e diferentes formas de sistemas de incentivo (SCOTT, 2016).

Como os preços de mercado das criptomoedas são baseados na oferta e na demanda, a taxa na qual uma criptomoeda pode ser trocada por outra moeda pode variar bastante, pois o projeto de muitas criptomoedas garante um alto grau de escassez (ULRICH, 2017). Há uma

preocupação de que criptomoedas como o Bitcoin não estejam enraizadas em quaisquer bens materiais. Algumas pesquisas, no entanto, identificaram que o custo de produção de um Bitcoin, que requer uma quantidade cada vez maior de energia, está diretamente relacionado ao seu preço de mercado.

Muitos investidores veem vantagens potenciais em criptomoedas, como a possibilidade de preservar valor contra a inflação e facilitar o câmbio, além de serem mais fáceis de transportar e dividir do que metais preciosos e existirem fora da influência de bancos centrais e governos (ULRICH, 2017).

3.3.1 O Bitcoin

O Bitcoin é uma moeda digital, não física, transacionada somente de forma eletrônica, que possui como característica principal a descentralização, sendo assim, não é controlada por nenhuma instituição monetária, sem vínculos com bancos e países. Tem origem recente, foi proposta em 2008, mas sua primeira moeda foi ser “minerada” apenas no ano de 2009.

Em “Bitcoin – a moeda na era digital”, Fernando Ulrich define o Bitcoin:

Bitcoin é uma moeda digital peer to peer (par a par ou, simplesmente, de ponto a ponto), de código aberto, que não depende de uma autoridade central. Entre muitas outras coisas, o que faz o Bitcoin ser único é o fato de ele ser o primeiro sistema de pagamentos global totalmente descentralizado. (ULRICH, 2017).

Apesar de ter sido criada em 2008, foi apenas em 2010 que a criptomoeda começou a ser usada de forma mais generalizada e crescente. A ideia inovadora da Bitcoin trouxe uma nova perspectiva ao que conhecemos atualmente como moeda. A sua complexidade e aceitação a nível mundial fazem dela um fenômeno interessante de ser estudado.

Atualmente, existem cerca de 3 milhões de bitcoins que ainda precisam ser extraídos. Dessa forma, o Bitcoin (e qualquer criptomoeda gerada por um processo semelhante) opera de forma diferente da moeda fiduciária, em sistemas bancários centralizados, a moeda é liberada a uma taxa correspondente ao crescimento de mercadorias, na tentativa de manter a estabilidade de preços, enquanto um sistema descentralizado como o Bitcoin define a taxa de liberação antecipadamente e de acordo com um algoritmo (PAAR; PELZL, 2009).

3.4 Séries temporais

Uma série temporal é uma sequência de dados atrelados a um certo intervalo de tempo. Se for observada uma variável por um conjunto de pontos no tempo e registrado seu comportamento, a variável formará uma tendência contra o tempo. Essa tendência é conhecida como série temporal (BRINK et al., 2017).

Uma série temporal pode ser observada em qualquer variável que muda ao longo do tempo. Em finanças, é comum usar uma série temporal para rastrear o preço de um título ao longo do tempo. Isso pode ser feito no curto prazo, como o preço de um título na hora ao longo de um dia útil, ou no longo prazo, como o preço de um título no último dia de cada ano (GURGEL; FORMIGA, 2013).

A análise de séries temporais pode ser aplicada a qualquer variável que muda ao longo do tempo e, de um modo geral, os pontos de dados mais próximos são mais semelhantes do que os mais distantes. Na maioria das vezes, os componentes dos dados de séries temporais incluem uma tendência, sazonalidade, ruído ou aleatoriedade, uma curva e o nível. É importante observar que nem todos os dados de séries temporais incluirão cada um desses componentes (COSTA, 2016).

Por exemplo, arquivos de áudio que são capturados em sequência e formam uma série temporal, no entanto, eles não contêm um componente sazonal (embora note que eles teriam ciclos periódicos). Por outro lado, a maioria dos dados comerciais provavelmente conterá sazonalidade, como o pico das vendas no varejo no quarto trimestre. A análise de séries temporais é uma técnica estatística que lida com dados de séries temporais ou análise de tendências. Dados de séries temporais significa que os dados estão em uma série de períodos ou intervalos específicos (SERAFIM; SILVA NETO, s.d.). Os dados são considerados em três tipos:

- Dados de séries temporais: um conjunto de observações sobre os valores que uma variável recebe em momentos diferentes.
- Dados agrupados: uma combinação de dados de séries temporais e dados de seção transversal.

Neste contexto, a modelagem e previsão de séries temporais estão se tornando cada vez mais populares em vários campos. No presente estudo, a série temporal formada a partir do registro de fechamento da cotação do Bitcoin servirá para a predição das cotações futuras.

Capítulo 4

Ferramentas e Métodos

Como apontado anteriormente, o principal objetivo dessa dissertação foi criar e aplicar um modelo de rede neural artificial para predição do preço do Bitcoin. Neste capítulo é descrita a metodologia e ferramentas utilizadas para implementação da proposta.

4.1 Seleção da criptomoeda

Para dar início ao processo de desenvolvimento dessa pesquisa, fez-se necessário definir o objeto principal de estudo. A criptomoeda escolhida atende aos critérios pré-estabelecidos (alta liquidez e volatilidade) que potencializa a capacidade de ganhos. Portanto, dentre as diversas opções de criptomoedas, escolheu-se o Bitcoin, por ser a primeira criptomoeda, a mais influente e com maior volume de negociação diária.

4.2 Linguagem de programação Python

Existem diversas linguagem de programação para o desenvolvimento de scripts, dentre elas, o Python que vem se destacando e crescendo constantemente com suas bibliotecas para análise e visualização de dados. São diversos os fatores que corroboram para difusão dessa linguagem, os principais são: diversidade de bibliotecas que apoiam desenvolvedores e cientistas à solucionarem seus problemas de maneira mais descomplicada e a possibilidade de ser executada em diversas plataformas. Como dito anteriormente, um dos elementos que determina a utilização da linguagem de programação Python é a diversidade de bibliotecas focadas em análise e visualização de dados. Tendo isso em mente e conhecendo a

comunidade que se formou em torno da linguagem, Python foi a linguagem de programação escolhida para auxiliar essa pesquisa.

4.3 Biblioteca Pandas

A biblioteca pandas é um pacote disponível na linguagem de programação Python que fornece fácil acesso a dados disponíveis na internet. O objetivo principal do Pandas é, justamente, fornecer buscas flexíveis e intuitivas para a análise de dados estruturados e não estruturados. Além disto, busca como objetivo específico, se tornar a ferramenta de análise, busca e manipulação de dados, aberta, mais poderosa e flexível existente.

4.4 Biblioteca Matplotlib

Matplotlib é uma biblioteca para para visualização de dados. Seu principal objetivo oferecer recursos para criação de gráficos a partir de um conjunto de dados. Assim como a biblioteca Pandas, citada anteriormente, o Matplotlib possui comandos simples e intuitivos, dispensando conhecimento aprofundado para sua utilização. O matplotlib é uma biblioteca disponível para a linguagem de programação Python que contém recursos para a geração de gráficos 2D, a partir de um determinado conjunto de dados. Seus gráficos são criados com alta qualidade utilizando comandos simples e intuitivos. (HUNTER, 2007) segmenta a biblioteca em da seguinte forma:

- `pylab`: conjunto de funções disponíveis em `matplotlib.pyplot` que permite a escrita de códigos simples de serem criados;
- `API`: camada responsável por realizar toda a lógica da biblioteca, criando por exemplo figuras, textos, gráficos e linhas;
- `backend`: conjunto de funções que dependem do dispositivo de saída, como por exemplo imagens e arquivos no formato Portable Document Format (PDF).

4.5 Biblioteca Sklearn

Sklearn é uma biblioteca da linguagem Python desenvolvida especificamente para aplicação prática de machine learning. Esta biblioteca dispõe de ferramentas simples e eficientes para

análise preditiva de dados, é reutilizável em diferentes situações, possui código aberto, sendo acessível a todos.

4.6 Especificação do modelo de rede neural artificial

A especificação de um modelo de rede neural artificial é de fato uma fase muito importante e com total influência no resultado final. É fundamental ressaltar que não existe um modelo pré-determinado e eficaz em RNAs, muitas das vezes é preciso analisar os resultados gerados, e, se necessário, utilizar o método de tentativa e erro para alcançar os resultados desejados. Entretanto, com o conhecimento adquirido no capítulo anterior, esse processo pode ser menos trabalhoso do que aparenta. Analisar o problema em que se está trabalhando, juntamente com o conhecimento teórico, é de grande valia para a especificação e construção de um modelo que se aproxime do resultado esperado. Para facilitar a especificação do modelo que será aplicado, ou seja, modelos que possuam características referentes a resolução do problema em questão, alguns sinais serão analisados, são eles:

1. Qual é o formato dos dados?

Essa pergunta aborda o modelo de transmissão dos dados que a RNA deve agir, ou seja, se a RNA será formada por camadas adiante ou recorrente. Visto que na nesta pesquisa serão analisados dados através de séries temporais, a RNA deverá trabalhar com um modelo recorrente.

2. O treinamento será realizado com base nos resultados esperados?

Essa pergunta aborda o processo de aprendizagem que deve ser utilizado. Se os dados são treinados com base nos resultados desejados, a RNA terá que possuir um método de aprendizado supervisionado, caso contrário, um modelo de RNA com treinamento não supervisionado terá que ser empregado. Nesse trabalho, a RNA será treinada com base nos resultados esperados através das séries temporais históricas, ou seja, o modelo trabalhará com aprendizagem supervisionada.

4.6.1 Definição da arquitetura

De acordo com os dois critérios estabelecidos na seção anterior, chegou-se a definição de que a RNA deverá possuir uma arquitetura de camadas recorrentes, um método de aprendizado supervisionado e uma função de custo, para controlar o processo de generalização.

Segundo (HAYKIN, 2007), o modelo Long Short-term Memory (LSTM) é indicado nesse caso, uma vez que a mesma possui a capacidade de aprender dependências de longo prazo.

Com a arquitetura da rede já definida, a próxima etapa foi definir as entradas que foram utilizadas na RNA. Para essa pesquisa foram elaborados dois algoritmos, o primeiro leva em consideração apenas o preço do fechamento do bitcoin para prever o preço nos dias subsequentes e o segundo leva em consideração diversas outras variáveis para a realização da previsão. A fim de facilitar a compreensão, seguem os parâmetros que compuseram a camada de entrada da RNA em cada um dos algoritmos:

1. Algoritmo com uma entrada

- *Close*: Valor de fechamento

2. Algoritmo com múltiplas entradas

- *Open*: Valor de abertura
- *High*: Valor máximo atingido
- *Low*: Valor mínimo atingido
- *Close*: Valor de fechamento
- *Adj Close*: Fechamento ajustado
- *Volume BTC*: Quantidade total negociada em Bitcoin

(HAYKIN, 2007) nos alerta para a importância da especificação da camada escondida, pois o número de neurônios depende de vários fatores. Os dois principais problemas nesse tipo de algoritmo são:

- *Overfitting*: Excesso de neurônios e camadas, levando a rede a memorizar os padrões de treinamento e quebrar seu processo de aprendizagem.
- *Underfitting*: Poucos neurônios para uma grande quantidade de dados, pode levar a rede a não encontrar uma solução desejável, pelo pouco poder de processamento.

Com isso, o modelo desenvolvido possui três camadas intermediárias compostas por 50 neurônios cada, buscando otimizar a sua estrutura e obtendo um resultado eficiente para o problema. Ainda como precaução de causar um *overfitting*, foi implementado uma técnica

conhecida como *Early Stopping*, que permite especificar um número grande arbitrário de épocas de treinamento e interromper o treinamento assim que o desempenho do modelo parar de melhorar, evitando assim o sobreajuste (Overfitting).

Finalmente, a saída da rede é composta por uma única camada que contém o valor de fechamento do dia seguinte. Na Figura 4.1 é possível visualizar a composição da arquitetura proposta.

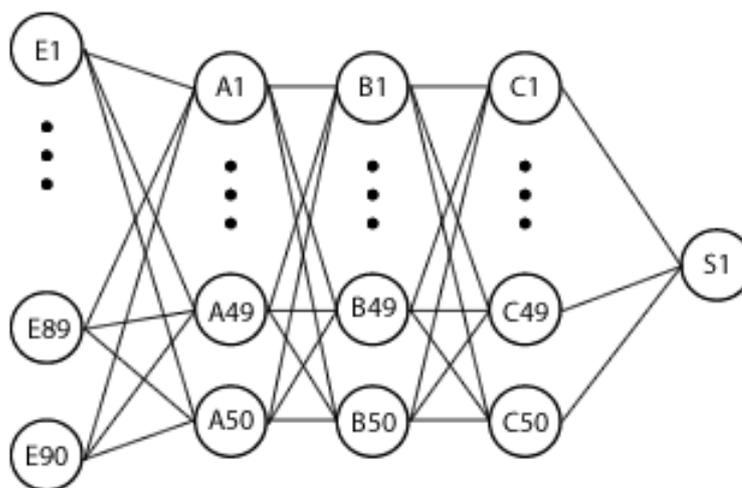


Figura 4.1: Representação do modelo de arquitetura proposta.
Fonte: Elaborado pelo autor

4.7 Configuração de Hardware

Toda a implementação das técnicas foram realizadas em um computador pessoal com um processador Intel Core i3-4200U com 3.20 gigahertz e uma memória RAM DDR3 com 6 gigabytes.

Desta forma, todos os métodos necessários para realizar a implementação deste trabalho foram levantados. O próximo capítulo irá detalhar o processo de desenvolvimento das técnicas, e, conseqüentemente, da obtenção dos objetivos desejados, levando todos os aspectos estudados e definidos até o momento.

Capítulo 5

Desenvolvimento

O propósito desta seção é, de forma mais detalhada, apresentar todo o processo de implementação realizado no corrente trabalho, com o intuito de atingir os objetivos determinados no Capítulo 1.

5.1 Coleta de dados

A base de dados utilizada nesse estudo foi extraída com o auxílio da plataforma Yahoo Finance¹ que disponibiliza dados do mercado financeiro e de diversas criptomoedas, dentre elas, o Bitcoin.

Os dados são disponibilizados no seguinte formato para todos valores diários:

- Date: Dia analisado
- Open: Valor de abertura
- High: Valor máximo atingido
- Low: Valor mínimo atingido
- Close: Valor de fechamento
- Adj Close: Fechamento ajustado
- Volume BTC: Quantidade total negociada em Bitcoin

¹<https://finance.yahoo.com/quote/BTC-USD/history?p=BTC-USD>

Atualmente temos aproximadamente 2064 registros nas base de dados, sendo o primeiro registro datado de 17/10/2014. O site disponibiliza o download dos dados no formato CSV². Já na obtenção da base de dados, utilizou-se a biblioteca Pandas para explorar o dataframe e prepara-lo para posterior utilização no modelo de rede neural artificial recorrente.

5.2 Pré-processamento

A etapa de pré-processamento é de suma importância e tem relação direta com a qualidade da predição. Nesta etapa foram feitos os tratamentos necessários para adaptar o dataframe original ao modelo aceito pela rede neural artificial. Portanto, esse processo foi dividido nas seguintes etapas:

1. Leitura da base de dados e remoção dos registros nulos (null). O método `dropna()` da biblioteca pandas nos auxilia nessa tarefa.

```
base = pd.read_csv('BTC.csv')
base = base.dropna()
```

Figura 5.1: Leitura do dataframe.
Fonte: Elaborado pelo autor

2. Seleção das colunas. O método `iloc` permite que selecionemos a coluna desejada e descartamos o restante.

- Seleção da coluna *Close* para o algoritmo 1.

```
base = base.iloc[:, 4:5].values
```

Figura 5.2: Seleção da coluna *Close*.
Fonte: Elaborado pelo autor

- Seleção das colunas *Open*, *High*, *Low*, *Close* e *Adj Close* para o algoritmo 2.

```
base_treinamento = base.iloc[:, 1:7].values
```

Figura 5.3: Seleção da coluna *Close*.
Fonte: Elaborado pelo autor

²CSV é um formato de arquivo que significa "valores separados por vírgulas". Isso significa que os campos de dados indicados neste formato normalmente são delimitados por vírgulas.

3. Normalização dos dados. A normalização consiste em transformar uma base de dados escalando cada registro para um determinado intervalo, evitando alguns poucos registros manipulem o resultado final, o que se aplica à realidade do presente trabalho, uma vez que o preço do Bitcoin conta com intervalos de grande oscilação e eventuais valores discrepantes. Para essa etapa, utilizou-se o método *MinMaxScaler()* da biblioteca Sklearn, com o intervalo de 0 a 1.

- Normalização da base de dados para o algoritmo 1.

```
normalizador = MinMaxScaler(feature_range=(0,1))
base = normalizador.fit_transform(base)
```

Figura 5.4: Normalização dos dados Normalização dos dados algoritmo 1.

Fonte: Elaborado pelo autor

- Normalização da base de dados para o algoritmo 2.

```
normalizador = MinMaxScaler(feature_range=(0,1))
base_treinamento_normalizada = normalizador.fit_transform(base_treinamento)

normalizador_previsao = MinMaxScaler(feature_range=(0,1))
normalizador_previsao.fit_transform(base_treinamento[:,0:1])
```

Figura 5.5: Normalização dos dados Normalização dos dados algoritmo 2.

Fonte: Elaborado pelo autor

4. Modelagem da base de dados. A etapa de modelagem da base de dados consistiu em preparar os dados de maneira a ficarem compatíveis com o esperado pela rede neural, a Figura 5.8 exemplifica o processo de transformação da base de dados.

- Modelagem da base de dados para o algoritmo 1.

```
periodo_ant = 90

previsores = []
preco_real = []
for i in range (periodo_ant, len(base)):
    previsores.append(base[i-90:i, 0])
    preco_real.append(base[i,0])
previsores, preco_real = np.array(previsores), np.array(preco_real)
previsores = np.reshape(previsores, (previsores.shape[0], previsores.shape[1],1))
```

Figura 5.6: Modelagem do dataframe para o algoritmo 1.

Fonte: Elaborado pelo autor

- Modelagem da base de dados para o algoritmo 2.

```

periodo_ant = 90

previsores = []
preco_real = []
for i in range (periodo_ant, len(base)):
    previsores.append(base_treinamento_normalizada[i-periodo_ant:i, 0:6])
    preco_real.append(base_treinamento_normalizada[i,0])
previsores, preco_real = np.array(previsores), np.array(preco_real)

```

Figura 5.7: Modelagem do dataframe para o algoritmo 2.

Fonte: Elaborado pelo autor

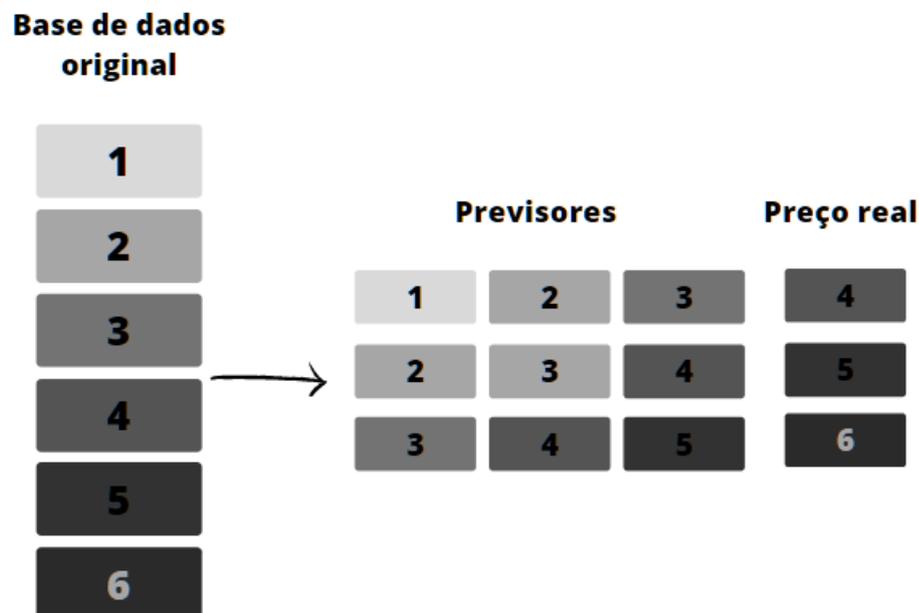


Figura 5.8: Criação da base de treinamento.

Fonte: Elaborado pelo autor

5. Divisão da base de dados em base de treinamento e base de teste. Utilizou-se a proporção 33% para treino e 67% para teste. (HAYKIN, 2000) propõe esta divisão, quando há uma grande quantidade de amostras disponíveis para o treinamento.

5.3 Desenvolvimento da rede neural artificial

A implementação de uma RNA foi construída através do módulo Keras.models. O mesmo tem como objetivo, especificar como são tratados o fluxo dos dados de entrada, processamento e saída da rede. A figura 5.9 demonstra a criação da topologia determinada.

```
regressor = Sequential()
regressor.add(LSTM(units = 90, return_sequences = True, input_shape = (previsores.shape[1],6)))
regressor.add(Dropout(0,5))

regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0,5))

regressor.add(LSTM(units = 50 , return_sequences = True))
regressor.add(Dropout(0,5))

regressor.add(LSTM(units = 50))
regressor.add(Dropout(0,5))

regressor.add(Dense(units = 1, activation = 'Linear'))
regressor.compile(optimizer = 'rmsprop', loss = 'mean_squared_error', metrics = ['mean_absolute_error'])

es = EarlyStopping(monitor = 'Loss', min_delta = 1e-10, patience = 10 , verbose = 1)
rlr = ReduceLROnPlateau(monitor = 'Loss', factor = 0.2, patience = 5, verbose = 1)
mcp = ModelCheckpoint(filepath = 'pesos.h5', monitor = 'Loss', save_best_only = True, verbose = 1)
```

Figura 5.9: Estrutura da rede neural.

Fonte: Elaborado pelo autor

Utilizou-se o otimizador RMSprop³ para os pesos e juntamente os métodos erro quadrático médio (mean squared error) e erro médio absoluto (mean absolute error) para avaliação da perda do modelo.

5.4 Análise dos Resultados

O propósito desta seção é apresentar os resultados alcançados com os algoritmos desenvolvidos no capítulo anterior.

Após uma sequência de fases, como, especificação do modelo, definição da arquitetura e desenvolvimento da RNA, é chegado o momento de avaliar o desempenho do modelo proposto, afim de mensurar a capacidade de prever a cotação futura do Bitcoin, principal objetivo desse trabalho.

5.4.1 Aplicação do primeiro algoritmo (1 entrada)

Esta RNA foi treinada tendo como entrada apenas a série temporal e tem como objetivo testar a capacidade de generalização do modelo com um única entrada. A métrica definida para análise do resultado foi o comportamento da função de custo que compõe o modelo (erro médio quadrado, MSE). A Figura 5.10 representa, de maneira ilustrativa, os resultados da rede.

³RMSprop é uma técnica de otimização baseada em gradiente usada no treinamento de redes neurais . Foi proposto por Geoffrey Hinton. De forma simplificada, o RMSprop usa uma taxa de aprendizado adaptativa em vez de tratar a taxa de aprendizado como um hiperparâmetro . Isso significa que a taxa de aprendizagem muda com o tempo.

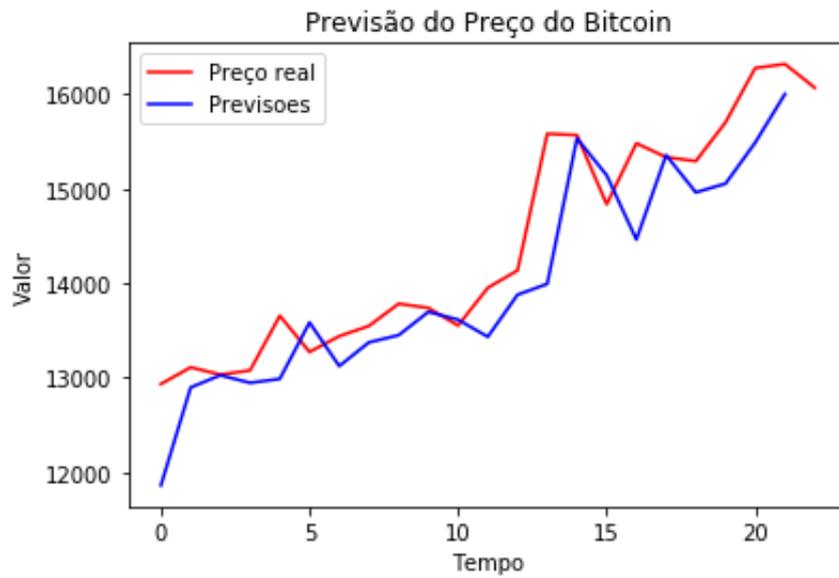


Figura 5.10: Curva de previsão algoritmo 1.
Fonte: Elaborado pelo autor

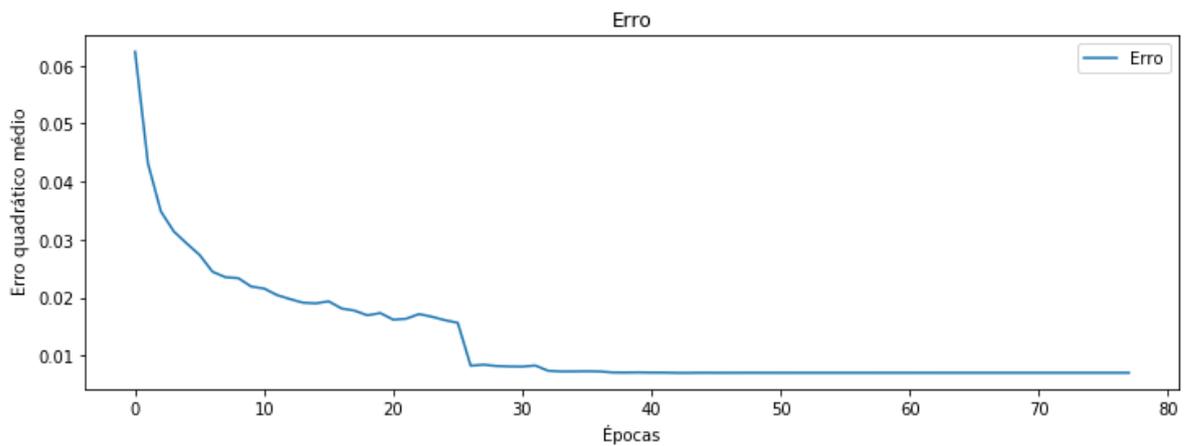


Figura 5.11: Decaimento do MSE no treinamento da rede.
Fonte: Elaborado pelo autor

Analisando a Figura 5.11, constata-se que houve uma queda significativa na taxa de erro, até 27 iterações. No intervalo de 28 a 76 iterações nota-se uma estabilização e a convergência acontece próximo a septuagésima oitava. Analisando o comportamento da curva, conclui-se que o modelo se comportou bem e de forma estável. Os resultados obtidos, a partir da execução dos testes, são demonstrados na Tabela 5.1.

Média Previsão	Média Preço Real	Erro Absoluto	Erro %
13.976,769	14.419,757	442.988	3,07

Tabela 5.1: Resultados da predição realizada pelo algoritmo 1.
Fonte: Elaborado pelo autor

5.4.2 Aplicação do segundo algoritmo (múltiplos previsores)

Esta RNA foi treinada tendo como entrada apenas a série temporal e tem como objetivo testar a capacidade de generalização do modelo com um único previsor. A métrica definidas para análise do resultado foi o comportamento da função de custo que compõem o modelo (erro médio quadrático, MSE). A Figura 5.12 representa, de maneira ilustrativa, os resultados da rede. Esse experimento foi realizado

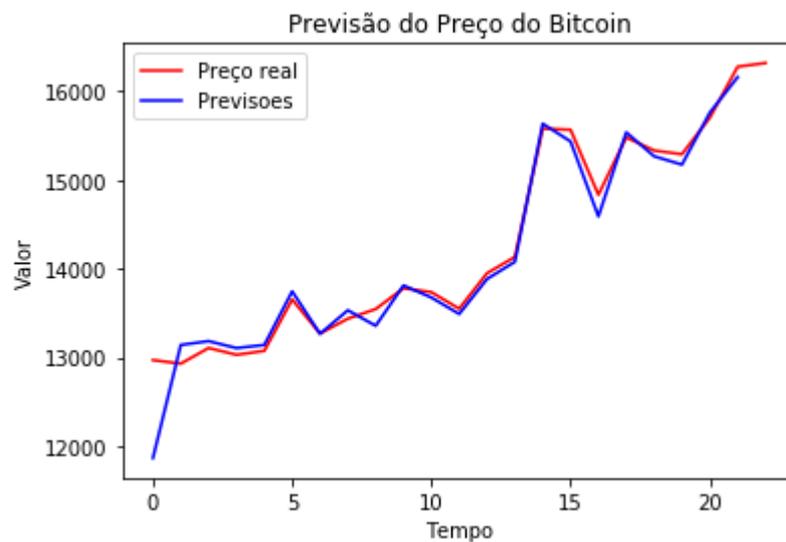


Figura 5.12: Curva de previsão algoritmo 2.
Fonte: Elaborado pelo autor

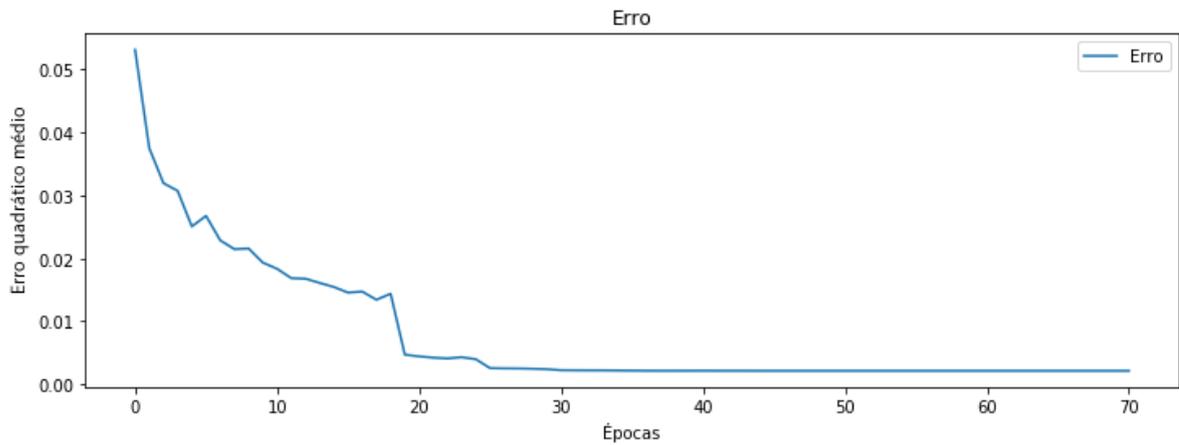


Figura 5.13: Decaimento do MSE no treinamento da rede.

Fonte: Elaborado pelo autor

Analisando a Figura 5.13, constata-se que houve uma queda significativa na taxa de erro, até 20 iterações. No intervalo de 20 a 69 iterações nota-se uma estabilização e a convergência acontece próximo a septuagésima iteração. Analisando o comportamento da curva, conclui-se que o modelo se comportou bem e de forma estável. Os resultados obtidos, a partir da execução dos testes, são demonstrados na Tabela 5.2.

Média Previsão	Média Preço Real	Erro Absoluto	Erro %
14.129,522	14.419,757	290,235	2,01

Tabela 5.2: Resultados da predição realizada pelo algoritmo 2.

Fonte: Elaborado pelo autor

Capítulo 6

Conclusões

O propósito dessa dissertação foi construir um modelo de rede neural artificial e aplicá-lo, a fim de prever a cotação futura da criptomoeda Bitcoin, ativo que se destaca por sua volatilidade e possibilidade de ganhos com sua valorização.

Para o desenvolvimento de uma RNA com resultados satisfatórios, foi necessário realizar pesquisas sobre conceitos fundamentais que envolvem a utilização desta técnica. No decorrer de tais pesquisas identificou-se questões essenciais que auxiliaram na concepção da rede, tais como arquiteturas de redes neurais e modelos de redes baseados em séries temporais. Desta forma, definiu-se que o modelo mais apropriado nesse caso seria o de redes neurais recorrentes, com arquitetura LSTM.

Analisando os resultados obtidos, observa-se que o modelo e a arquitetura proposta da RNA obtiveram resultados satisfatórios, alcançando assim, o principal objetivo do trabalho. Evidenciando esses resultados, observou-se que o resultado da rede, nos dois casos estudados, não ultrapassou a margem de 3,07% de erro em relação ao valor real.

Por fim, prever uma variável relacionada ao preço dada a multiplicidade de forças que impactam o mercado, não é fácil. Acrescente a isso, o fato de que os preços são, em grande maioria, especulação de perspectivas futuras e não dependem de dados históricos. No entanto, o uso de redes neurais profundas, nos forneceu com uma melhor compreensão do Bitcoin e da arquitetura LSTM.

É interessante destacar, como recomendação para trabalhos futuros, o emprego de redes neurais que atuam com séries temporais associadas a outras técnicas de inteligência artificial, tais como algoritmos genéticos e análise de sentimento de texto, proporcionando uma arquitetura híbrida para a tentativa de predições.

Também recomenda-se avaliar outras topologias e arquiteturas de redes neurais, fomentando uma arquitetura ideal para o problema proposto. Em relação a modelagem dos dados, é interessante realizar uma pesquisa mais detalhada sobre fatores internos e externos que influenciam na cotação do bitcoin, alimentando assim a rede com informações mais relevantes e correlacionadas.

Referências bibliográficas

- AGUIAR, F. G. **Utilização de Redes Neurais Artificiais para detecção de padrões de vazamento em dutos**. 2010. Tese (Doutorado) – Universidade de São Paulo.
- ALESSANDRETTI, L.; ELBAHRAWY, A.; AIELLO, L. M.; BARONCHELLI, A. Anticipating cryptocurrency prices using machine learning. **Complexity**, Hindawi, v. 2018, 2018.
- ALPAYDIN, E. **Introduction to machine learning, 2nd edn. adaptive computation and machine learning**. [S.l.]: The MIT Press (February 2010), 2010.
- BRINK, H.; RICHARDS, J. W.; FETHEROLF, M.; CRONIN, B. **Real-world machine learning**. [S.l.]: Manning Shelter Island, NY, 2017.
- COSTA, R. C. Modelos Preditivos de Velocidade de Vento para Sistemas Eólicos Baseados em Redes Neurais Artificiais. **Trabalho de Conclusão de Curso. Universidade Federal da Paraíba**, 2016.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A.; BENGIO, Y. **Deep learning**. [S.l.]: MIT press Cambridge, 2016. v. 1.
- GUPTA. **Blockchain for Dummies IBM Limited Edition**. 1. ed. [S.l.]: John Wiley Sons, Inc, 2017. p. 128. ISBN 978-1-119-37139-7.
- GURGEL, S. T.; FORMIGA, A. d. A. Parallel implementation of feedforward neural networks on gpus. In: IEEE. 2013 Brazilian Conference on Intelligent Systems. [S.l.: s.n.], 2013. p. 143–149.
- HAYKIN, S. **Redes Neurais: princípios e prática. 2ª edição. Trad: Paulo Martins Engel**. [S.l.]: São Paulo: Bookman companhia editora, 2000.
- HAYKIN, S. **Redes neurais: princípios e prática**. [S.l.]: Bookman Editora, 2007.
- HENDRIK WITTE, J. The Blockchain: A Gentle Four Page Introduction. **arXiv**, arxiv-1612, 2016.
- HUNTER, J. D. Matplotlib: A 2D graphics environment. **Computing in science & engineering**, IEEE Computer Society, v. 9, n. 3, p. 90–95, 2007.
- KHANDELWAL, I.; ADHIKARI, R.; VERMA, G. Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition. **Procedia Computer Science**, Elsevier, v. 48, n. 1, p. 173–179, 2015.
- MCNALLY, S.; ROCHE, J.; CATON, S. Predicting the price of bitcoin using machine learning. In: IEEE. 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP). [S.l.: s.n.], 2018. p. 339–343.

PAAR, C.; PELZL, J. **Understanding cryptography: a textbook for students and practitioners**. [S.l.]: Springer Science & Business Media, 2009.

PAULA, M. B. d. et al. Indução automática de árvores de decisão. Florianópolis, SC, 2002.

SAAD, M. et al. Toward characterizing blockchain-based cryptocurrencies for highly accurate predictions. **IEEE Systems Journal**, IEEE, v. 14, n. 1, p. 321–332, 2019.

SCOTT, B. **How can cryptocurrency and blockchain technology play a role in building social and solidarity finance**. [S.l.], 2016.

SERAFIM, D. C.; SILVA NETO, A. J. da. Estruturando Redes Neurais Artificiais Paralelas e Independentes para o Controle de Próteses Robóticas.

SILVA, A. J. da; OLIVEIRA, W. R. de; LUDERMIR, T. B. Weightless neural network parameters and architecture selection in a quantum computer. **Neurocomputing**, Elsevier, v. 183, p. 13–22, 2016.

ULRICH, F. **Bitcoin - A Moeda Na Era Digital**. Sao Paulo: Instituto Ludwig von Mises Brasil: LVM EDITORA, 2017. p. 123. ISBN 9788581190754. Disponível em: <<https://books.google.com.br/books?id=KblhtAEACAAJ>>.