

**UNIVERSIDADE FEDERAL DE ALAGOAS**

**INSTITUTO DE COMPUTAÇÃO**

**COORDENAÇÃO DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**PROPOSTA DE DISSERTAÇÃO DE MESTRADO**

**MODELOS DE PREVISÃO DE ATIVOS UTILIZANDO  
APRENDIZAGEM PROFUNDA E LINGUAGEM NATURAL**

**MESTRANDO(A)**

**JOÃO VICTOR DE LIMA MOURA**

**ORIENTADOR(A)**

**XU YANG**

**MACEIÓ, AL**

**AGOSTO - 2020**

**JOÃO VICTOR DE LIMA MOURA**

**ORIENTADOR(A)**

**XU YANG**

**MACEIÓ, AL**  
**AGOSTO - 2020**

**Catálogo na fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**  
**Divisão de Tratamento Técnico**

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 - 1767

M929m Moura, João Victor de Lima.

Modelos de previsão de ativos utilizando aprendizagem profunda e linguagem natural / João Victor de Lima Moura. – 2020.  
50 f. : il.

Orientadora: Xu Yang.

Dissertação (mestrado em Informática) - Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2020.

Bibliografia: f. 47-50.

1. Modelo de sentimento (Inteligência artificial). 2. Bolsa de valores. 3. Redes neurais (Computação). 4. Aprendizado profundo. 5. Análise de séries temporais. I. Título.

CDU: 004.81:159.942.52



## Folha de Aprovação

João Victor de Lima Moura

### “MODELO DE PREDIÇÃO DA BOLSA DE VALORES BASEADO EM DEEP LEARNING E MINERAÇÃO DE DADOS”

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal de Alagoas e aprovada em 27 de agosto de 2020.

#### Banca Examinadora:

Xu Yang

---

**Professora Dra. Xu Yang**  
UFAL – Instituto de Computação

C

---

**Prof. Dr. Evandro de Barros Costa**  
UFAL – Instituto de Computação  
**Examinador Interno**

Yuri F. Saporito

---

**Prof. Dr. Yuri Fahham Saporito**  
FUNDAÇÃO GETULIO VARGAS- Escola de Matemática Aplicada  
**Examinador Externo**

# AGRADECIMENTOS

Neste espaço gostaria de agradecer a todos que diretamente e indiretamente contribuíram para a realização desta pesquisa.

Agradeço a minha família, Gabrielly de Lima Moura, minha irmã, Rosineide Mendonça Moura, minha tia, João Mendonça Moura, meu pai, por todo o incentivo, paciência e colaboração durante todo o período.

Gostaria de agradecer ao José Carlos dos Santos Freitas pela paciência, noites mal dormidas ouvindo minhas reclamações, angustias, pela ajuda no desenvolvimento desta tese, por não permitir desistir e seguir até o fim.

As duas amigas maravilhosas que foram encontradas no Mestrado Ially Cristina Silveira De Almeida e Andressa Carvalho Melo da Silveira, por fazerem a caminhada ser mais fácil.

A minha orientadora Professora Dr. Xu Yang por toda paciência comigo durante todo o período.

A amigos cultivados desde a primeira graduação, engenharia ambiental, em especial à Bárbara Karoline Soares Fernandes Dantas, que mesmo distante ofereceu todo seu apoio por telefone.

À Fundação de Amparo a Pesquisa do Estado de Alagoas (FAPEAL) por todo o apoio financeiro durante estes 24 meses.

E por fim, a pessoa mais importante, minha falecida mãe, que sempre lutou para me propiciar uma excelente educação, que me mostrou que o caminho do sucesso se passa pelos estudos.

A todos, meu muito obrigado!

## Resumo

Predizer o futuro é algo que a humanidade anseia, de modo a poder se preparar para possíveis eventualidades. Hoje, aliando a facilidade de acesso a informações através de notícias com o uso de programas que “leem” os sentimentos dos usuários na internet, é possível usar a linguagem computacional para prever, inclusive, o comportamento do mercado financeiro. No âmbito da economia, essa predição possibilita novas formas de intervir no mercado, propiciando o vislumbre de diversos cenários, facilitando desse modo a tomada de decisão sobre investimentos e sobre o futuro econômico de empresários, empresas e governos. Neste sentido, as ferramentas de inteligência artificial vêm se tornando uma boa ponte entre as técnicas de previsão e as técnicas de engenharia, úteis na referida abordagem, apresentando tratamentos quantitativos e análises de limitações de mercado. O presente trabalho visa desenvolver um modelo de predição da bolsa de valores Brasileira, elaborado na linguagem de programação Python, e verificar a sua capacidade preditiva através da submissão dos dados das séries históricas em redes neurais Long Short Term Memory (LSTM) e Deep Neural Network (DNN). Para isto, foi utilizada mineração de notícias no Twitter de jornais confiáveis, sendo que os dados oriundos dos classificadores baseados em Naive Bayes e Support Vector Machine (SVM) foram incorporado as séries históricas das empresas Banco do Brasil e Petrobras para serem utilizados nas redes neurais supracitadas. Os resultados mostraram que os classificadores tiveram bons resultados, principalmente o classificador SVM, apresentando uma precisão de 82% de notícias classificadas corretamente para a Petrobras. Já as redes neurais apresentaram uma razoável capacidade preditiva, apresentando RSME de 26,4% utilizando a rede DNN, e 35,1% na rede LSTM para a Petrobras e RMSE de 32,70% utilizando a rede LSTM e 24,25% na rede DNN para o Banco do Brasil. Apesar do banco de dados utilizado ter sido relativamente pequeno, o que pode ter influenciado nos resultados deste trabalho, pesquisas apontam que, em geral, é possível relacionar notícias a alta e a baixa de ações. Quanto a isto, ainda há muitas questões a serem elucidadas a fim de serem encontradas previsões mais realísticas.

**Palavras-Chave:** Análise de Sentimento, Bolsa de Valores, Redes Neurais, Deep Learning, Séries Temporais.

## Abstract

Predicting the future is something that humanity craves for, in order to be able to prepare for possible eventualities. Today, combining facility of access to information through news with the use of programs that "read" the feelings of users on the Internet, it is possible to use computer language to even predict market behavior. In the scope of economy, this prediction enables new ways of intervening in the economic market, providing a glimpse of different scenarios, therefore facilitating the decision-making about investments and also about the economic future of entrepreneurs, companies and governments. In this sense, artificial intelligence tools have become a great bridge between forecasting techniques and engineering techniques, useful in that approach, presenting quantitative treatments and limitations analysis. The present work aims to develop a prediction model for the Brazilian Stock Exchange, developed in the Python programming language, and to verify its predictive capacity through the submission of historical series data in Long Short Term Memory (LSTM) and Deep Neural Network (DNN). For this, news were researched on Twitter from trusted newspapers, and data from classifiers based on Naive Bayes and Support Vector Machine (SVM) were incorporated into the historical series of the companies Banco do Brasil and Petrobras to be used in the aforementioned neural networks. The results showed that the classifiers had good results, mainly the SVM classifier, presenting an accuracy of 82% of correctly classified news for Petrobras. The neural networks had a reasonable predictive capacity, with RSME of 26.4% using the DNN network, and 35.1% in the LSTM network for Petrobras and RMSE de 32,70% using a the LSTM network e 24,25% using DNN network for the Banco do Brasil. Although the database used was relatively small, which may have influenced the results of this study, papers show that, in general, it is possible to relate news with the rising or falling of the stock market. In this regard, there are still many questions to be clarified in order to find more realistic predictions.

**Keywords:** Sentiment Analysis, Stock Exchange, Neural Networks, Deep Learning, Time Series.

## Lista de Figuras

1	Hierarquia conforme os tipos de aprendizado de maquina. Fonte: [1] . . . . .	6
2	Dendritos, corpo celular e axônio são os elementos que formam um neurônio simplificado, sua estrutura e funcionamento são a base de inspiração do paradigma das redes neurais artificiais. Fonte: [1] (p. 109) . . . . .	8
3	Modelo de uma RNAs multicamadas tradicional. Fonte: [1] (p. 281) . . . . .	9
4	Modelo de uma RNAs básica, configurada pelo ponto de vista matemático. Fonte: O autor . . . . .	12
5	Algoritmo de treinamento perceptron. Fonte: [1] . . . . .	14
6	Algoritmo de treinamento de uma rede backpropagation. Fonte: [1] . . . . .	14
7	Uma rede neural densamente conectada com duas camadas ocultas. Fonte: O autor . . . . .	15
8	Representação de uma separação de um SVM . . . . .	23
9	Gráficos das previsões de ativos para a Petrobras (A) e para o Banco do Brasil (B) na rede DNN para o período de 10 anos. Fonte: Yahoo Finance (2020), copilado pelo autor . . . . .	37
10	Gráficos das previsões de ativos para a Petrobras (A) e para o Banco do Brasil (B) na rede LSTM para o período de 10 anos. Fonte: Yahoo Finance (2020), copilado pelo autor . . . . .	38
11	Gráficos das previsões de ativos para a Petrobras (A) e do Banco do Brasil (B) na rede LSTM com adição da análise de sentimento para o período de 9 meses. Fonte: Yahoo Finance (2020), copilado pelo autor. . . . .	39
12	Gráficos das previsões de ativos para a Petrobras (A) e do Banco do Brasil (B) na rede DNN com adição da análise de sentimento para o período de 9 meses. Fonte: Yahoo Finance (2020), copilado pelo autor. . . . .	40



## Lista de Tabelas

1	As quatro principais regras de aprendizagem que definem como a rede deve ajustar os pesos sinápticos. . . . .	10
2	Matriz de confusão do classificador de Naive Bayes para as empresas Petrobras e Banco do Brasil . . . . .	35
3	Matriz de confusão do classificador de SVM para Petrobras e Banco do Brasil	35
4	Comparativo das medidas de quantificação dos classificadores Naive Bayes e SVM . . . . .	36
5	Hiper parametros para as ações da Petrobras e Banco do Brasil da rede DNN	37
6	Hiper parametros para as ações da Petrobras e Banco do Brasil da rede LSTM	38
7	Hiper parametros para as ações da Petrobras e Banco do Brasil da rede LSTM com análise de sentimento . . . . .	39
8	Hiper parametros para as ações da Petrobras e Banco do Brasil da rede DNN com análise de sentimento . . . . .	40
9	Comparativo dos RMSE das redes neurais LSTM e DNN . . . . .	40

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Fundamentação Teórica</b>	<b>4</b>
2.1	Machine learning	4
2.1.1	Algoritmo <i>Naive Bayes</i>	6
2.1.2	Redes Neurais Artificiais	7
2.2	<i>Deep Learning</i>	14
2.3	Tipos de Camada Importantes	15
2.3.1	Camada Densa	16
2.3.2	<i>Dropout</i>	16
2.4	Rede Neural Recorrente	16
2.5	Memória de Longo Prazo	17
2.6	Trabalhos Relacionados	17
2.6.1	Usando codificadores automáticos empilhados (SAEs) e memória de curto prazo (LSTM)	18
2.6.2	Integrando a abordagem de mineração de texto usando notícias em tempo real	18
2.6.3	Previsão do preço das ações usando regressão linear com base na análise de sentimentos	19
2.7	Limitações dos modelos existentes	19
<b>3</b>	<b>Análise de Sentimento</b>	<b>20</b>
3.0.1	Classificação Técnica	21
3.0.2	<i>Support Vector Machine (SVM)</i>	21
3.0.3	<i>Naïve Bayes</i>	23
3.0.4	Entropia máxima	24
3.0.5	Sentença e Level	24
3.0.6	Trabalhos Relacionados à análise de sentimentos	25
<b>4</b>	<b>Metodologia</b>	<b>29</b>
4.1	Extração de dados	29

4.1.1	<i>Twitter</i>	29
4.1.2	Yahoo Finances	30
4.2	Extração de dados	30
4.2.1	Tratamento de dados do Twitter	30
4.2.2	Tratamento de dados do Twitter	30
4.3	Classificador	31
4.3.1	Acúrcia	31
4.3.2	Precisão	32
4.3.3	<i>Recall</i>	32
4.3.4	<i>F-Measure</i>	32
4.4	Redes LSTM	32
4.5	<i>Deep FeedForward Neural Network (DNN)</i>	33
4.6	Validação	33
<b>5</b>	<b>Resultados e Discussão</b>	<b>34</b>
5.1	Resultado dos classificadores	34
5.1.1	Comportamento do <i>Naive Bayes</i>	34
5.1.2	Comportamento do SVM	35
5.2	Predição sem análise de sentimentos	36
5.2.1	Rede DNN	37
5.2.2	Rede LSTM	37
5.3	Predição com análise de sentimentos	38
5.4	Discussão	41
<b>6</b>	<b>Conclusão</b>	<b>44</b>

# 1 Introdução

O preço das ações é o preço de uma única ação entre o número de ações vendidas por uma empresa listada em oferta pública. Ter ações de uma empresa pública permite que você seja dono de uma parte dela. Os proprietários originais da empresa vendem inicialmente as ações para obter investimentos adicionais para ajudar a empresa a crescer. Essa oferta inicial de ações ao público é chamada de *Initial Public Offering* (IPO).

Os preços das ações mudam devido à oferta e demanda. Suponha que, se muitas pessoas estão dispostas a comprar uma ação, o preço sobe à medida que há mais demanda. Se mais pessoas estão dispostas a vender as ações, o preço cai, pois há mais oferta do que demanda. Embora o entendimento da oferta e da demanda seja relativamente fácil, é difícil derivar quais fatores contribuem exatamente para o aumento da demanda ou da oferta. Esses fatores geralmente se resumem a variáveis socioeconômicas, como comportamento de mercado; inflação; tendências e, mais importante, as notícias vinculadas a empresa na grande mídia.

Atualmente, graças ao avanço da ciência e tecnologia da informação, é possível criar simuladores que podem auxiliar na tomada de decisão. Ramo fecundo dentro da Tecnologia da Informação, o uso de algoritmos que “leem” as informações e conseguem processar dados de forma ágil e imediata chama a atenção para a sua aplicabilidade em diversas áreas, tais como a Economia.

Aliando Tecnologia da Informação com a Economia é possível dá os primeiros passos para um dos maiores desejos do ser humano: a predição do futuro. Mas aqui não se está falando sobre as predições baseadas em astrologia ou karma, mas sim, no comportamento de determinados fatores que podem favorecer o estabelecimento de uma linealidade do comportamento, sendo este positivo ou negativo.

No âmbito da Bolsa de Valores, os investidores tem buscado desde os seus primórdios formas e técnicas que favoreçam no melhor retorno financeiro, uma vez que nas Bolsas grandes somas de capitais são investidos todos os dias, com o intuito, seguindo a lógica do capital, de gerar lucros. Aos investidores que conseguem divisar os melhores cenários e situações, lhes serão concedidos maiores cenários favoráveis com o retorno de lucro esperado e efetivo.

Para a tomada de decisão dos melhores cenários e situações que favorecem o investi-

mento e retorno de lucros, pautar-se em análises técnicas são imprescindíveis, sendo elas: gráficos, tendências de mercado, índices da bolsa e informações de jornais e microblogs conceituados e de referência nacional. Entretanto, o grande número de informações que estão disponíveis para análise, sob a ótica humana é quase impossível de ser processada e garantir uma decisão imediata. Portanto, obter um ambiente que consiga prever tendências de modo autônomo e eficaz são imperiosos para a Economia, para tanto alguns pesquisadores vem fazendo uso de Inteligência Artificial (IA) para analisar essas informações e assim auxiliar na tomada de decisão

Algumas das primeiras pesquisas em previsão de preços de ações datam de 1994, em que foi realizado um estudo comparativo [2] com modelos de regressão de aprendizado de máquina. Desde então, graças aos avanços da ciência, muitos pesquisadores investiram recursos para elaborar estratégias para aperfeiçoar prever o preço das ações, criando cenários novos de análise, incluindo novos paradigmas.

A previsão do valor futuro de ações é do interesse de investidores, que procuram utilizar diversos indicadores para fazer suas previsões. Atualmente, diversos trabalhos procuram algoritmos capazes de prever o valor das ações com base na análise dos sentimentos dos usuários de redes sociais, expressos em textos das redes sociais. A pesquisa destes algoritmos é importante para os referidos profissionais, que podem tomar decisões mais acertadas, tendo como base um índice referente ao sentimento de um conjunto significativo de pessoas, que expressam os seus sentimentos nas redes sociais.

Os especuladores da bolsa de valores são importantes para gerar maior liquidez no mercado. Graças a esta liquidez, o preço de diversas *commodities* como arroz, feijão, soja, boi e etc são estabilizados durante o ano. Antes de haver estas negociações, o valor do arroz, por exemplo, era consideravelmente elevado no período da entressafra deste alimento. Por conta disso, ao desenvolver um algoritmo que torna a operação nas bolsas de valores mais atrativa, contribui-se com a sociedade inteira, devido ao aumento da liquidez que se segue ao maior número de negociadores de *commodities*, por exemplo (benefícios análogos se dão no caso das ações de empresas e outros papéis negociados na bolsa de valores).

Este trabalho tem como objetivo pesquisar as formas existentes de previsão do comportamento do preço das ações do mercado financeiro através da análise do sentimento dos usuários nas redes sociais. Neste sentido, os objetivos específicos são:

- Pesquisar na literatura recente as técnicas e algoritmos utilizados na análise de sentimento de textos presentes nas redes sociais.
- Desenvolver sistema de análise de sentimentos de textos do twitter capaz de prever o movimento futuro dos preços nas bolsas de valores.

Com relação a estrutura da presente monografia, objetivou-se no capítulo quatro trazer a fundamentação teorica que dará suporte científico e metodológico para o presente trabalho, onde buscou-se abordar a historia e conceito de *machine learning*, onde foram tecidas considerações, dando enfase nos modelos supervisionados e não supervisionados; conceituação sobre redes neurais e sobre *deep learning*.

Já no capítulo cinco, foi exposto o que é análise de sentimentos e sua aplicabilidade na tematica deste trabalho. Além disto, abordou-se a importância dos classificadores (*Support Vector Machine - SVM* e *Naive Bayes*) no tratamento das informações.

Por sua vez, o capítulo seis, versa sobre a metodologia utilizada neste trabalho, onde expõe sobre a aplicação dos classificadores na analise de dados do Twitter e como foram utilizados os modelos de redes neurais para realizar a predição dos ativos.

No penultimo capítulo estão expostos os resultados e discussão do trabalho, onde é possível identificar os pontos em comuns e divergentes do presente trabalho.

No último capítulo estão tecidas as considerações finais deste trabalho, evidenciando a aplicabilidade do uso de redes neurais e classificadores de sentimentos para a predição de ativos na bolsa de valores.

## 2 Fundamentação Teórica

Este capítulo apresenta a fundamentação teórica do trabalho. Expõe os princípios de machine learning e seus principais algoritmos utilizados, bem como o big data que forma os pilares que permitem a aplicação dos algoritmos de machine learning atualmente incluindo a evolução do processamento de dados. Tal como o cenário atual quanto ao uso de aplicações em destaque para problemas reais e conceitos que possam fornecer melhor entendimento do trabalho.

### 2.1 Machine learning

Machine learning ou aprendizado de máquina é uma subárea da inteligência artificial a qual, consiste na correlação com outras áreas, entre elas a probabilidade e estatística; teoria da computação; neurociência e correlatas. Nos primórdios da era da computação, a inteligência artificial era vista como algo fictício e de pouca aplicabilidade, cenário esse que tem mudado drasticamente nos últimos anos.

A aceitação e desenvolvimento é caracterizada no instante em que se começa produzir soluções para problemas reais, o gênese da inteligência artificial neste escopo é dada pelos sistemas especialistas [3], baseada na aquisição do conhecimento de um domínio, entretanto tal modalidade era muito dependente de um especialista humano

Desta forma vê-se a necessidade de ferramentas mais autônomas, que aprendam a partir de experiências passadas e que sejam capazes de criar uma função aproximada para resolver problemas a partir de dados

As técnicas de *Machine learning* ganharam destaque a partir do momento em que se tornou possível a utilização de recursos computacionais mais eficazes e eficientes. Entre os recursos pode-se citar o Big Data, impulsionando o crescente volume de dados disponíveis, o processamento computacional mais acessível e eficiente. De acordo com Rezende (2003) [4] o aprendizado de máquina pode ser classificado quanto ao critério de aprendizado de tarefas, sendo:

- Supervisionado (preditivo)
- Não Supervisionado (descritivo)

O autor Soares et. al. [5] caracteriza o aprendizado supervisionado de máquina:

No aprendizado supervisionado é fornecido ao algoritmo de aprendizado, ou indutor, um conjunto de exemplos de treinamento para os quais o rótulo da classe associada é conhecido. Em geral, cada exemplo é descrito por um vetor de valores de características, ou atributos, e pelo rótulo da classe associada. O objetivo do algoritmo de indução é construir um classificador que possa determinar corretamente a classe de novos exemplos ainda não rotulados, ou seja, exemplos que não tenham o rótulo da classe. Para rótulos de classe discretos, esse problema é conhecido como classificação e para valores contínuos como regressão.

Complementarmente os autores John Muellere Luca Massaron[6] caracterizam o aprendizado de máquina não supervisionado:

A aprendizado não supervisionado ocorre quando um algoritmo aprende a partir de exemplos claros, sem nenhuma resposta associada, deixando o algoritmo determinar os padrões de dados por conta própria. Esse tipo de algoritmo tende a reestruturar os dados em algo diferente, como novas características que representam uma classe ou uma nova série de valores não correlacionados. São muito úteis para dar ideias às pessoas sobre o significado dos dados e novas entradas para algoritmos de aprendizado de máquina supervisionados. Como tipo de aprendizado, se assemelham aos métodos utilizados pelos seres humanos para descobrir que certos objetos ou eventos pertencem à mesma classe, como ao observar o grau de semelhança entre eles.

A Figura 1 apresenta uma hierarquia conforme os tipos de aprendizado já definidos:

Conforme Faceli et al. [1],pode-se organizar os algoritmos de machine learning quanto:

- Supervisionado
  - *k-Nearest Neighbor*
  - *Naive Bayes*
  - Árvores de Decisão e Regressão





Figura 1: Hierarquia conforme os tipos de aprendizado de máquina. Fonte: [1]

- Máquinas de vetores de suporte
- Redes Neurais Artificiais
- Supervisionado
  - Variantes do algoritmo K-médias
  - Variantes de Redes Neurais(*Deep Learning*)
  - Baseados em grafos
  - Baseados em grid

Nas próximas seções serão especificados alguns dos principais algoritmos de machine learning em evidência.

### 2.1.1 Algoritmo *Naive Bayes*

De acordo com Anasse Bari, Mohamed Chaouchie Tommy Jung [7] “algoritmo *Naive Bayes* é baseado no Teorema de Bayes, uma poderosa fórmula que pode ser usada para calcular a probabilidade de um evento com base em conhecimento prévio”.

Conforme asseguram Bianchi e Thiago [8], a Teoria de *Bayes* mostra a relação entre a probabilidade condicional e sua inversa, ou seja a probabilidade de uma hipótese dada a observação de uma evidência e a probabilidade da evidência dada pela hipótese. Eles definem a evidência como sendo o conhecimento prévio e frequentemente relacionado ao termo probabilidade. O Teorema de Bayes demonstrado na equação (1) declara:

$$Prob(X|Y) = [P(X)P(Y|X)]/[P(X)P(Y|X) + P(\neg X)P(Y|\neg X)] \quad (1)$$

onde:

- $P(X)$  é a probabilidade de o Evento  $X$  ocorrer.
- $P(\neg X)$  é a probabilidade de o Evento  $X$  não ocorrer.
- $P(X|Y)$  é a probabilidade de o Evento  $X$  ocorrer, considerando que o Evento  $Y$  ocorreu.
- $P(Y|X)$  é a probabilidade de o Evento  $Y$  ocorrer, considerando que o Evento  $X$  ocorreu.
- $P(Y|\neg X)$  é a probabilidade de o Evento  $Y$  ocorrer, considerando que o Evento  $X$  não ocorreu.

### 2.1.2 Redes Neurais Artificiais

Um algoritmo de redes neurais artificiais tem sua origem em processos observados nos sistemas nervosos biológicos, ou seja na estrutura do cérebro humano, a maioria das redes neurais fazem uso de funções matemáticas, com a capacidade de produzir uma saída para cada uma das entradas. As redes neurais são identificadas segundo Kovács e Zsolt László [9] como:

uma classe de modelos matemáticos para problemas de classificação e reconhecimento de padrões outras ainda como uma parte da teoria conexionista dos processos mentais e finalmente, como uma categoria de modelos em ciência da cognição.

A estrutura do neurônio dos seres vivos, ao qual as redes neurais artificiais são inspiradas, tem fator de importância uma vez que sua concepção proporciona a base das atividades deste paradigma, seu esquema básico é visto na Figura 2 [1].

Os dendritos tem a função de receber os estímulos de outros neurônios, o corpo por sua vez processa e combina as informações e direciona o impulso para o axônio; o axônio então

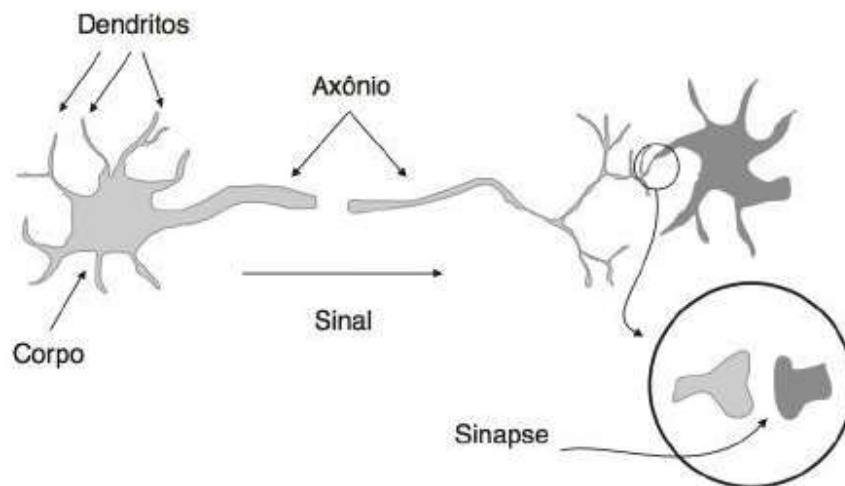


Figura 2: Dendritos, corpo celular e axônio são os elementos que formam um neurônio simplificado, sua estrutura e funcionamento são a base de inspiração do paradigma das redes neurais artificiais. Fonte: [1] (p. 109)

estende o sinal do neurônio na ordem da esquerda para direita tendo como destino as sinapses nervosas; sendo as sinapses junções comunicantes entre os neurônios, podendo ser de dois tipos excitatórios ou inibitórios.

Nos modelos computacionais, combinando os neurônios em uma ou mais camadas que podem ter uma ou mais neurônios interligando esses neurônios através das sinapses, pode-se formar o que se denomina redes neurais artificiais. A principal característica de uma rede neural é a sua capacidade de aprender a partir de um ambiente e de melhorar seu desempenho por meio desta aprendizagem, podendo ser dividido em duas formas: aprendizado supervisionado e aprendizado não supervisionado.

Na aprendizagem supervisionada a rede é treinada com pares de conjunto de entradas e de saídas, quando é apresentado a rede em conjunto de entrada, esta retorna um conjunto de valores de saída que é comparado ao conjunto de valores de saída desejado. Aquilo que foi observado (os dados históricos), em seguida os pesos sinápticos em níveis de baias são ajustados de forma minimizar a diferença entre a saída apresentada pela rede e pela saída desejada. Esse processo é repetido para todos os pares de entrada e saída que constituem o conjunto de treinamento da rede até que a taxa de acerto seja considerada satisfatória.

Na aprendizagem não supervisionada a rede não recebe informações sobre a saída dese-

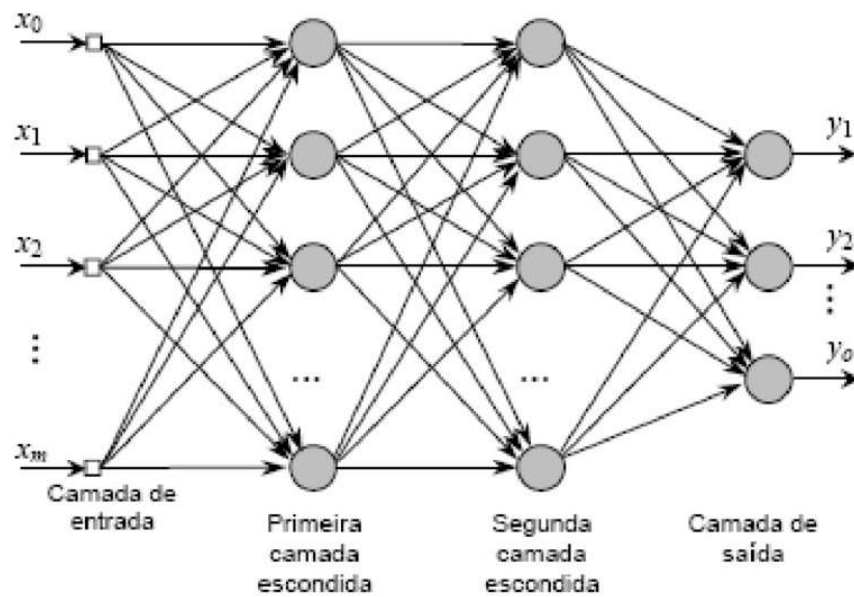


Figura 3: Modelo de uma RNAs multicamadas tradicional. Fonte: [1] (p. 281)

jada, a rede é treinada apenas com valores de entrada e organiza sua estrutura de modo a fazer a classificação desses valores em grupos. Este tipo de rede tem por finalidade a classificação de dados pelo reconhecimento de padrões, através da detecção de características.

Todos os processos de aprendizagem ocorrem na primeira camadas dos neurônios, responsável pela entrada ou de distribuição (que tem esse nome apenas como figurativo pois não possui nenhum neurônio), tendo apenas o número de nós igual ao número de sinais de entrada da rede, a única função é representar a distribuição dos dados de entrada para cada neurônio da camada seguinte, não sendo realizado qualquer computação nesta camada. Na camada seguinte pode conter nenhuma, uma ou várias camadas ocultas, que são constituídas de um ou mais neurônios.

A função destes neurônios é capacitar a rede a extrair estatísticas, além de possibilitar a representação de problemas que não sejam linearmente separáveis. Finalmente têm-se uma camada de saída a qual contém necessariamente o número de neurônios igual a quantidade de sinais da rede. Ajustando os valores e seus pesos sinápticos a rede neural pode representar ou memorizar as relações entre os dados de entrada e saída, assumindo uma característica de memória associativa.

A atualização da representação é realizada sobre a forma de modificação da arquitetura, ajuste dos pesos das conexões entre os neurônios e ativando regras de neurônios individuais.

As regras de aprendizagem definem como a rede deve ajustar os pesos sinápticos (que é o grande segredo quando se trabalha com redes neurais). Existem 4 tipos principais de regras de aprendizado, resumidas na Tabela 1.

Tipo de regra de aprendizagem	Descrição
Aprendizagem por Correção e Erro	Utilizado em treinamento supervisionado, esta técnica ajusta os pesos sinápticos por meio do erro, que é obtido através da diferença entre o valor de saída da rede e o valor esperado em um ciclo de treinamento. Com isso gradualmente vai diminuindo o erro geral da rede
Aprendizagem Hebbiana	Baseado no postulado de aprendizagem de Hebb, que afirma: "se dois neurônios em ambos os lados de uma sinapse são ativados sincronamente e simultaneamente, então a força daquela sinapse é seletivamente aumentada". Este processo de treinamento é feito localmente, ajustando o peso das conexões baseado nas atividades dos neurônios.
Aprendizagem competitiva	Neste modelo os neurônios de uma rede de camada única competem entre si para permanecerem ativos, de modo que o neurônio que obtiver o maior nível, oriundo da multiplicação dos seus vetores de peso e vetores de entrada. Neste caso, diferente do modelo de Hebbiano, apenas um neurônio de saída fica ativo por vez.
Aprendizagem de Boltzmann	Método de aprendizagem estocástico derivado de conceitos da Estatística. Neste modelo os neurônios são estocásticos, podendo residir em dois estados possíveis, ligado (+1) e desligado (-1), e ainda são divididos em dois grupos funcionais, presos e livres, sendo responsáveis pela interação com o ambiente e pela explicação das restrições subjacentes dos padrões de entrada do ambiente, respectivamente.  Um ponto importante deste modelo de aprendizagem é que os neurônios possuem conexões bidirecionais.

Tabela 1: As quatro principais regras de aprendizagem que definem como a rede deve ajustar os pesos sinápticos.

A partir da estrutura e funcionamento do neurônio biológico, pesquisadores tentaram simular esse sistema em computador (sendo a primeira tentativa em 1945) (Figura 4). Neste modelo os impulsos elétricos provenientes de outros neurônios são representados pelos chamados sinais de entrada ( $X$ ) do diagrama dentre os vários estímulos recebidos. Assim como no modelo biológico os estímulos podem ser excitatórios ou inibitórios, representado pelo peso sináptico positivo ou negativo respectivamente. Alguns excitarão mais ou menos o neurônio receptor, e essa medida quando excitatório (estímulo), é representado pelo modelo McCulloch.

Quanto maior o valor do peso, mais excitatório é o estímulo. Os pesos sinápticos são representados por  $W_{KN}$  no diagrama, onde  $K$  representa o índice do neurônio em questão e,  $N$  se refere ao terminal de entrada da sinapse, ao qual o peso sináptico faz alusão. A soma do corpo da célula é representado pela composição de dois módulos: a junção aditiva, configurado como o somatório dos estímulos que são os sinais de entrada multiplicado pelo fator excitatório, os quais são os pesos sinápticos e, posteriormente por uma função de ativação que definirá com base nas entradas e nos pesos sinápticos qual será a saída do neurônio, o axônio é representado pela saída  $YK$  obtido pela aplicação da função de ativação. O modelo proposto possui uma natureza binária, uma vez que os valores de entrada e saída são binários.

O neurônio matemático, similarmente ao biológico, recebe um ou mais sinais de entrada e devolve um único sinal de saída que pode ser distribuído como o sinal da rede ou como sinal de entrada para um ou mais neurônios de camadas posteriores, os sinais de entrada chegam simultaneamente aos neurônios, podendo a rede ser classificada como instrumento de processamento paralelo.

Em máquinas sequenciais esse processo paralelo é simulado pela execução de uma varredura através da rede, entretanto é possível realizar esta mesma execução em paralelo utilizando GPUs (unidade de processamento gráfico). Os dendritos e os axônios são representados matematicamente apenas pelas sinapses e a intensidade da ligação é representada pela grandeza peso sináptico, simbolizada pela letra  $Y$ . Quando as entradas  $X$  são apresentadas ao neurônio elas são multiplicadas pelos pesos sinápticos correspondentes gerando as entradas ponderadas ou seja:  $X_1W_{k1}, X_2W_{k2}, \dots, X_mW_{km}$ .

O neurônio então totaliza todos os produtos e o resultado deste somatório é denominado por  $V$ , como função de combinação. Este valor é então apresentado a uma função de ati-

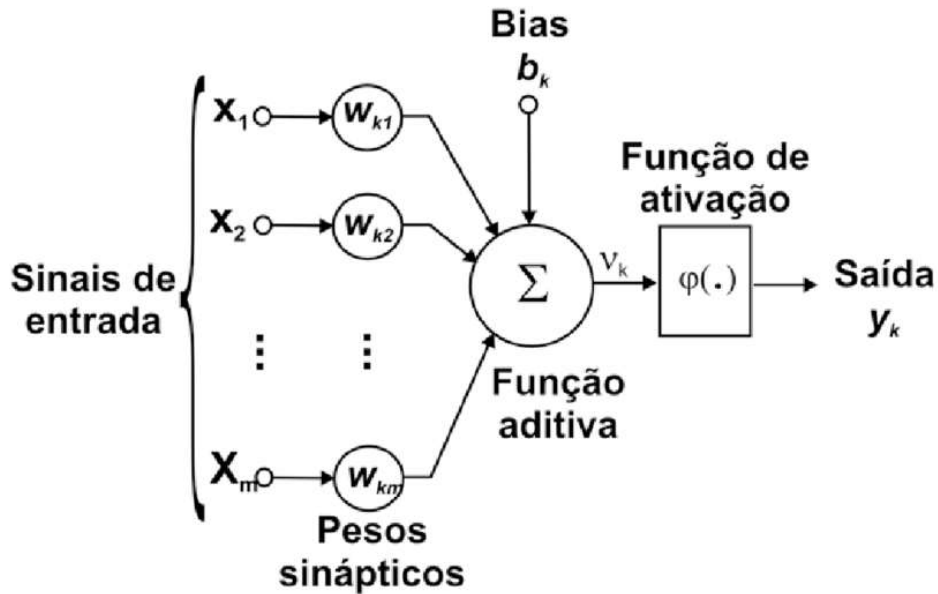


Figura 4: Modelo de uma RNAs básica, configurada pelo ponto de vista matemático. Fonte: O autor

vação, ou função de transferência, que tem a finalidade de evitar o acréscimo progressivo dos valores de saída ao longo das camadas da rede, visto que tais funções possuem valores máximos e mínimos contidos em intervalos determinados.

O uso de transferência não linear torna a rede neural uma feramente poderosa. O corpo de um neurônio é emulado por mecanismos simples que fazem a soma dos valores  $x_1W_{k1}, x_2W_{k2}, \dots, x_mW_{km}$  recebidos como soma ponderada e decide se deve ou não disparar, isto é sair igual à 0 ou 1, comparando a soma linear obtida do neurônio.

A ativação do neurônio é obtida através de uma função de ativação, que pode culminar em um resultado ou não, dependendo da soma ponderada das suas entradas. Observa-se que ainda não houve êxito em reproduzir exatamente o que o cérebro faz. Em termos matemáticos o neurônio artificial pode ser representado por essa primeira formula:

$$k = \sum_{j=0}^m xW_{kj} \cdot X_j. \quad (2)$$

onde,  $m$  é o número de sinais de entrada incidentes no neurônio  $k$ , e posteriormente aplicamos a função de ativação:

$$Y_k = \phi(U_k) \quad (3)$$

De acordo com [1] a função de ativação recebe o valor provido pela junção aditiva (soma-

tório) denominado por campo local induzido; atribuindo 1 se o campo local induzido daquele neurônio for não negativo e 0 caso contrario conforme a expressão:

$$\phi = 1 \text{ se } u \leq 0 \quad (4)$$

$$\phi = 0 \text{ se } u \geq 0 \quad (5)$$

Outras funções de ativação são apresentadas na literatura entre elas a função sigmoide:

$$\phi(v) = 1 \text{ se } u \geq 0 \quad (6)$$

$$\phi(v) = 0 \text{ se } u < 0 \quad (7)$$

Sendo mais utilizada em funções de redes neurais artificiais e tem o comportamento estritamente crescente que exhibe um balanceamento entre o comportamento linear e não linear. Mas outras funções de ativação podem ser utilizadas tais como a função da tangente hiperbólica:

$$\phi(v) = \tanh(v) \quad (8)$$

O valor das bias<sup>1</sup> é ajustado da mesma forma que os pesos sinápticos, a bia permite que o neurônio apresente saída não nula ainda que todas as suas entradas sejam nulas. Caso não houvesse o bias e todas as entradas fossem nulas então o valor da função de ativação seria nula.

Frank Rosenblatt propôs o primeiro modelo de implementação baseado em aprendizado supervisionado de redes neurais, a chamada rede Perceptron de camada única, usada para classificação de padrões linearmente separáveis. Entretanto muitos problemas não podiam ser resolvidos com apenas uma camada, surge então o algoritmo backpropagation permitindo o treinamento em mais de uma camada.

Os algoritmos de treinamento de uma rede perceptron (Figura 5 e Figura 6) e backpropagation são descritos na sequenciaria das imagens.

---

<sup>1</sup>Bias é um parâmetro adicional na Rede Neural que é usado para ajustar a saída junto da soma ponderada das entradas para o neurônio. Ou seja, Bias é uma constante que ajuda o modelo para poder ter uma melhor adaptação aos dados fornecidos.



**Entrada:** Um conjunto de  $n$  objetos de treinamento  
**Saída:** Rede perceptron com valores dos pesos ajustados

```

1 Inicializar pesos da rede com valores baixos
2 repita
3   para cada objeto  $x_i$  do conjunto de treinamento faça
4     Calcular valor da saída produzida pelo neurônio,  $\hat{f}(x_i)$ 
5     Calcular erro =  $y_i - \hat{f}(x_i)$ 
6     se erro > 0 então
7       Ajustar pesos do neurônio utilizando Equação 7.2
8     fim
9   fim
10 até erro = 0;
```

Figura 5: Algoritmo de treinamento perceptron. Fonte: [1]

**Entrada:** Um conjunto de  $n$  objetos de treinamento  
**Saída:** Rede MLP com valores dos pesos ajustados

```

1 Inicializar pesos da rede com valores aleatórios
2 Inicializar  $erro_{total} = 0$ 
3 repita
4   para cada objeto  $x_i$  do conjunto de treinamento faça
5     para cada camada da rede, a partir da primeira camada intermediária faça
6       para cada neurônio  $n_{ji}$  da camada atual faça
7         Calcular valor da saída produzida pelo neurônio,  $\hat{f}$ 
8       fim
9     fim
10    Calcular  $erro_{parcial} = y - \hat{f}$ 
11    para cada camada da rede, a partir da camada de saída faça
12      para cada neurônio  $n_{ji}$  da camada atual faça
13        Ajustar pesos do neurônio utilizando Equação 7.3
14      fim
15    fim
16    Calcular  $erro_{total} = erro_{total} + erro_{parcial}$ 
17  fim
18 até  $erro_{total} < \xi$ ;
```

Figura 6: Algoritmo de treinamento de uma rede backpropagation. Fonte: [1]

## 2.2 Deep Learning

Deep learning é um termo ambíguo, pois passou por vários significados diferentes ao longo dos anos, podendo ser definido como a decomposição de conceitos complexos em conceitos simples e a recombinação em novos conceitos complexos. Um algoritmo, portanto, tem que estabelecer uma hierarquia de conceitos, onde a visualização desta hierarquia seria um gráfico multi-camadas, que pode ser chamado de “profundo” (*deep* em inglês) em um contexto de teoria de grafos.

Neste trabalho, o termo será usado em um sentido mais restrito, como proposto por

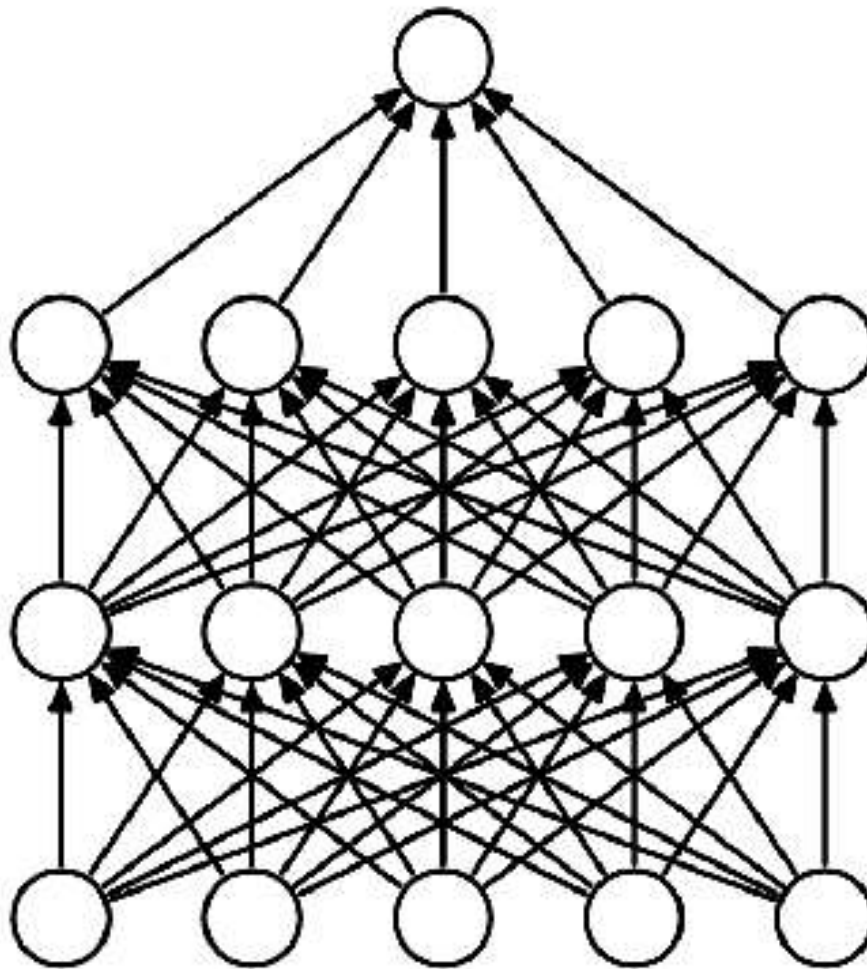


Figura 7: Uma rede neural densamente conectada com duas camadas ocultas. Fonte: O autor

Skansi e Sandro [10], ao qual refere-se a redes neurais artificiais profundas como um sub-campo do aprendizado de máquina. Neste capítulo, os conceitos cruciais para entender a aprendizagem profunda são fornecidos, seguindo a descrição em Bodlaender et al. [11].

### 2.3 Tipos de Camada Importantes

Ainda que Krizhevsky et al.[12] tenham utilizado três camadas diferentes na sua pesquisa, por estarem fazendo análise de imagens, no presente trabalho, pautou-se apenas na camada densa. Uma vez que este tipo de camada é a que melhor se comporta para a análise de dados numéricos. Além disso, a camada de abandono é introduzida porque provou ser um importante meio de regularização.

### 2.3.1 Camada Densa

A camada densa (Figura 7) usa o conjunto linear de conexões entre entrada e saída. Todas as unidades de uma camada densa são totalmente conectadas a todas as unidades nas camadas vizinhas, de tal forma que as unidades tomam cada saída de todas as unidades na antiga camada densa como sua entrada e passam sua saída para todas as unidades na camada subsequente.

Isso leva a uma computação de matriz vetorial simples, porém leva a um conjunto muito grande de parâmetros treináveis. Uma rede neural que consiste em camadas densas podem ser muito bem-sucedida para dados de baixa dimensionalidade, mas a computação pode se tornar muito cara para dados de alta dimensão, como imagens, para auxiliar com essas tarefas, as camadas com compartilhamento de peso são introduzidas.

### 2.3.2 Dropout

*Dropout* é um método para regularizar algoritmos de aprendizado de máquina. A idéia principal de uma camada de desistência é desabilitar aleatoriamente as unidades de entrada após cada interação do treinamento. Uma rede neural com  $n$  unidades que emprega o dropout pode ser vista como uma coleção de  $2n$  redes neurais possíveis. Essas redes possíveis têm um número menor de unidades, mas ainda compartilham pesos de forma que o número total de parâmetros não seja alterado. Formação de uma rede com abandono pode ser visto como a formação de um conjunto de  $2n$  menores redes com ampla partilha de peso [13].

## 2.4 Rede Neural Recorrente

Rede Neural Recorrente (RNN) é uma classe de RNA na qual as conexões entre os neurônios formam um grafo direcionado ou tendo um auto-loop nas camadas ocultas. Isso ajuda as RNNs a utilizar o estado anterior dos neurônios ocultos para aprender o estado atual. Juntamente com o exemplo de entrada atual, RNNs pegam as informações que aprenderam anteriormente no tempo. Eles usam estado interno ou memória para aprender informações sequenciais. Isso permite que eles aprendam uma variedade de tarefas, como reconhecimento de manuscrito, de fala entre outros.

Em uma rede neural tradicional, todas as unidades de vetores de entrada são consideradas

independentes. Como resultado, informações sequenciais não podem ser utilizadas na rede neural tradicional. Porém, enquanto no modelo RNN, os dados sequenciais da série temporal geram um estado oculto e são adicionados à saída da rede dependente do estado oculto. Como as tendências de estoque são um exemplo de dados de séries temporais, esse caso de uso é o mais adequado para RNNs.

Com base no resultado da camada de saída, as atualizações nos pesos das camadas ocultas são propagadas de volta. Mas em redes neurais profundas, essa mudança será pequena para as camadas no início, impedindo que os pesos alterem seu valor e interrompendo a rede para aprender mais. Isso é chamado de problema de gradiente de fuga.

## 2.5 Memória de Longo Prazo

A memória de longo prazo (LSTM, acrônimo em inglês para *Long Short-Term Memory*) é um tipo de arquitetura de rede neural recorrente na qual o problema do gradiente de fuga é resolvido. Os LSTMs são capazes de aprender dependências de longo prazo e funcionam tremendamente bem em uma grande variedade de problemas. LSTMs são introduzidos pela primeira vez por Hochreiter e Sepp[14]. Além dos autores originais, muitos pesquisadores contribuíram para a arquitetura das células LSTM modernas.

As redes neurais recorrentes geralmente são projetadas em uma estrutura semelhante a uma cadeia, retornando às camadas anteriores. Nos RNNs padrão, esse módulo de loop terá uma estrutura muito simples. Essa estrutura pode ser uma camada tanh simples que controla o fluxo.

Enquanto nos LSTMs, em vez dessa estrutura simples, eles têm quatro camadas de rede interagindo de maneira especial. Normalmente, os LSTMs são aumentados por portas chamadas "esquecer". Ao controlar essas portas, os erros podem ser retropropagados por qualquer número de camadas virtuais. Esse mecanismo permite que a rede aprenda tarefas que dependem de eventos que ocorreram várias vezes antes.

## 2.6 Trabalhos Relacionados

Trabalhos como Abe et. al.[15] provaram que é possível prever o movimento do preço das ações com base em técnicas de *deep learning*, fazendo uso de redes neurais artificiais.

Os autores conseguiram com o uso de Deep Neural Network, prever a movimentação das ações da bolsa de valores japonesa um mês antes, através da análise de 25 indicadores, os quais podem ser captados junto a bolsa japonesa.

Um número considerável de pesquisadores publicou trabalhos na última década sobre diferentes estratégias. Foram consideradas duas abordagens interessantes que tratam da mineração de informações de notícias e previsão de preço futuro de ações. Essas abordagens são explicadas brevemente nas seções a seguir [15].

### **2.6.1 Usando codificadores automáticos empilhados (SAEs) e memória de curto prazo (LSTM)**

Bao et. al.[16] apresentaram uma nova estrutura de *Deep Learning*, onde codificadores automáticos empilhados, transformações de longo prazo e *wavelet* (WT) são usados juntos para a previsão do preço das ações. Os SAEs para os recursos profundos que são extraídos hierarquicamente são introduzidos na previsão do preço das ações neste documento pela primeira vez.

Essa estrutura de *Deep Learning* consiste em 3 etapas. Em primeiro lugar, a WT decompõe a série temporal do preço das ações para eliminar o ruído. Em seguida, para geração de recursos de alto nível, os SAEs são aplicados para a previsão do preço das ações. Por fim, os recursos de denoising de alto nível são inseridos na memória de longo prazo para prever o preço de fechamento do dia seguinte.

### **2.6.2 Integrando a abordagem de mineração de texto usando notícias em tempo real**

As notícias são um fator muito importante que afeta os preços das ações. Um artigo positivo sobre o aumento das vendas de uma empresa pode se correlacionar diretamente com a volatilidade no preço das ações e vice-versa. Uma nova abordagem para extrair texto de notícias em tempo real e, assim, prever os preços das ações foi proposta no artigo de Kharde et. al. [17]. A mineração de informações textuais dos documentos e séries temporais simultaneamente é um tópico para aguçar o interesse na comunidade de mineração de dados. Há um aumento consistente no número de pesquisas realizadas nesta área [18, 19].

### 2.6.3 Previsão do preço das ações usando regressão linear com base na análise de sentimentos

Em 2018, Cakra et. al.[20] tentaram utilizar o sentimento das postagens de sites de mídia social, como o Twitter, para prever os preços das ações na Bolsa da Indonésia. Eles usaram *Naïves Bayes*, SVM e Randon Florest para classificar tweets sobre empresas e compararam os resultados dos diferentes algoritmos. Os autores alegaram que o modelo de *Random florest* alcançou o melhor desempenho entre os três algoritmos com precisão de 60,39%. *Naïve Bayes* ficou como o segundo melhor modelo com precisão de 56,50%. Em seguida, eles usaram algoritmos de classificação supervisionados, como SVM, Árvores de Decisão e regressão linear como modelos preditivos, e tentaram prever a flutuação de preços e o percentual de margem. Foi realizada uma análise comparativa dos resultados de todos os modelos [20].

## 2.7 Limitações dos modelos existentes

Embora a maioria das pesquisas anteriores nesse campo tenha se concentrado em técnicas para prever o preço das ações com base nos dados numéricos históricos, como tendências de ações passadas, não há muita pesquisa colocada na análise textual. Notícias e mídia têm enorme influência sobre os seres humanos e as decisões que tomam. Além disso, flutuações na bolsa de valores são resultado das atividades comerciais de seres humanos, sofrendo indiretamente influencia das notícias. Portanto, extrair informações de artigos de notícias pode produzir melhores resultados na previsão dos preços das ações. Previsão de tendências de ações sensíveis às notícias [19], análise de polaridade de sentimentos usando uma abordagem baseada em coesão [21], mineração de texto simultâneo e séries temporais [22] são alguns dos trabalhos destacados no domínio.

### 3 Análise de Sentimento

A análise de sentimentos também é conhecida como “mineração de opinião” ou *Emotion Artificial Inteligente* e alude à utilização do *processing natural language* (PNL), mineração de texto, linguística computacional e medições biológicas para reconhecer, extrair, avaliar e examinar metodicamente estados emocionais e informações subjetivas. A análise de sentimentos geralmente se preocupa com a voz nos materiais do cliente, tais como pesquisas e análises na Web e em redes sociais.

Como regra, a análise de sentimentos tenta determinar a disposição de um palestrante, ensaísta ou outros assuntos em termos de tema por meio de respostas emocionais ou apaixonadas extremas a um arquivo, comunicação ou ocasião. A disposição pode ser um julgamento ou avaliação, cheio de emoção (em outras palavras, a condição apaixonada do criador ou do orador) ou uma expectativa de respostas entusiasmadas (em outras palavras, o impacto pretendido pelo criador ou comprador).

Atualmente, existem muitas pesquisas ou recomendações de clientes sobre todos os tópicos disponíveis na Web, e as auditorias podem conter pesquisas sobre itens como clientes ou críticos de filmes. As pesquisas estão se expandindo rapidamente, considerando o fato das pessoas gostarem de expor suas opiniões na Web.

Para o pesquisador, ter acesso a diversas análises sobre um produto, pode prejudicar a sua análise pessoal ou empírica, pois ele passa a levar em consideração as informações disponibilizada por outros para formar a sua própria opinião. A mineração de sentimentos é uma tarefa que tira proveito das abordagens de PNL e extração de informações (IE) para analisar um grande número de arquivos, a fim de reunir os sentimentos de comentários apresentados por diferentes autores [23, 24]. Esse processo incorpora várias estratégias, incluindo etimologia computacional e recuperação de informações (RI).

A idéia básica da investigação de sentimentos é detectar a polaridade dos documentos de texto ou frases curtas e classificá-los nessa premissa. A polaridade do sentimento é categorizada como "positiva", "negativa" ou "imparcial"(neutra). É importante destacar o fato de que a mineração de sentimentos pode ser realizada em três níveis:

- Classificação de sentimento no nível do documento: nesse nível, um documento pode ser classificado inteiramente como "positivo", "negativo" ou "neutro".

- Classificação do sentimento no nível da sentença: nesse nível, cada sentença é classificada como "positiva", "negativa" ou imparcial.
- Classificação de sentimentos em nível de aspecto e característica: nesse nível, as sentenças / documentos podem ser categorizados como "positivo", "negativo" ou "não partidário" à luz de certos aspectos das frases / arquivos e comumente conhecido como "avaliação em nível de perspectiva" agrupamento".

### 3.0.1 Classificação Técnica

No campo de aprendizado de máquina, foram desenvolvidos métodos de classificação que usam estratégias diferentes para classificar dados não rotulados, onde classificadores podem exigir dados de treinamento. Exemplos de classificadores de aprendizado de máquina são *Naïve Bayes*, Entropia Máxima e SVM. Eles são classificados como métodos de aprendizado de máquina supervisionado. É importante mencionar que o treinamento eficaz de um classificador facilitará as previsões futuras.

### 3.0.2 *Support Vector Machine (SVM)*

O *Support Vector Machine (SVM)* foi analisada pela primeira vez em 1992, introduzida por Boser, Guyon e Vapnik na COLT-92. O SVM é uma ferramenta de previsão de classificação e regressão que usa a teoria de aprendizado de máquina para maximizar a precisão da previsão, evitando automaticamente o ajuste excessivo aos dados. O SVM foi inicialmente popular na comunidade *Neural Information Process System* e agora é parte ativa da pesquisa de aprendizado de máquina em todo o mundo, tendo ficado famoso ao usar mapas de pixel como entrada; ele fornece precisão comparável a redes neurais sofisticadas com recursos elaborados em uma tarefa de reconhecimento de manuscrito.

Ele também está sendo usado como análise de escrita manual e análise de face, por exemplo, especialmente para classificação de padrões e aplicativos baseados em regressão. As fundações de SVM foram desenvolvidas por Vapnik [25] e ganhou popularidade devido a muitas características promissoras, como melhor performance empírica. A fórmula utiliza o princípio da *Structural Risk Minimization (SRM)*, que tem se mostrado superior ao princípio tradicional de *Empiric Risk Minimization (ERM)*, usando por redes neurais convencionais. As



formulas de SRM e ERM estão listadas abaixo:

$$R_{emp}(h) = \frac{1}{m} \sum L(h(x_i), y_i) \quad (9)$$

$$R_{srn}(f) = \frac{1}{N} \sum L(y_i, f(x_i)) + \lambda J(f) \quad (10)$$

onde:

- $x$  e  $y$  representa, respectivamente, o conjunto de dados de treinamento e seu rótulo correspondente ;
- $L$  função para medir a perda;
- $\lambda J(f)$  é um regularizador de complexidade, geralmente pode ser o limite do espaço vetorial.

O SRM minimiza um limite superior no risco esperado, enquanto o ERM minimiza o erro nos dados de treinamento. É essa diferença que equipa a SVM com maior capacidade de generalização, que é o objetivo na aprendizagem estatística. SVMs foram desenvolvidos para resolver o problema de classificação, mas recentemente eles foram estendidos para resolver problemas de regressão.

O SVM também apresenta bom desempenho na análise de sentimentos, investigando informações, caracterizando os limites de escolha e usando os componentes para o cálculo, que são executados no espaço de entrada. A informação principal é apresentada em dois arranjos de vetores, cada qual com um determinado tamanho. Nesse ponto, cada dado (expresso como um vetor) é ordenado em uma classe. Em seguida, a máquina identifica o limite entre as duas classes que está longe de qualquer lugar nas amostras de treinamento. O separado caracteriza a borda da classificação, expandindo-a, o que diminui as opções ambivalentes. Como demonstrado em Khairnar et al. [26], foi comprovado que o SVM tem desempenho mais eficaz que o classificador Naïve Bayes em vários problemas de classificação de texto.

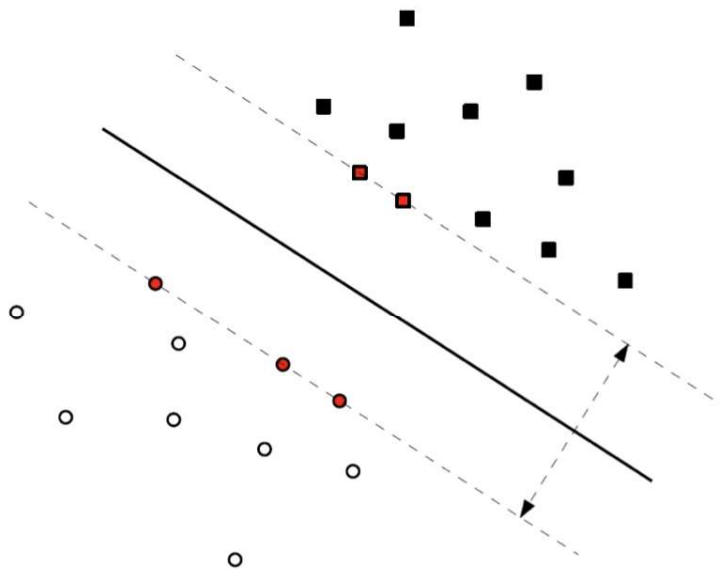


Figura 8: Representação de uma separação de um SVM

### 3.0.3 *Naïve Bayes*

Este é um método de classificação que se baseia no teorema de Bayes com fortes pressupostos de independência entre os recursos. Um classificador *Naïve Bayes* espera que a proximidade de um recurso específico (elemento) de uma classe seja desconectada da proximidade de alguns outros elementos. Por exemplo, uma fruta orgânica pode ser considerada uma maçã se sua cor é vermelha, sua forma for redonda e medir aproximadamente três polegadas de largura. Independentemente dessas características serem dependentes umas das outras ou da presença de outras características, um classificador *Naïve Bayes* consideraria essas propriedades independentes devido à probabilidade de que essa fruta natural seja uma maçã.

Sabe-se que a *Naïve Bayes* supera até as estratégias de ordem extremamente modernas. A hipótese de Bayes é um método de computação para distinguir a probabilidade  $P(a|b)$  de  $P(a)$ ,  $P(b)$  e  $P(b|a)$  da seguinte maneira:

$$p(a|b) = [p(b|a) * p(a)]/p(b) \quad (11)$$

onde  $p(a|b)$  está a probabilidade posterior da classe  $a$  dado preditor  $b$  e  $p(b|a)$  é a probabilidade que é a probabilidade do preditor  $b$  dada a classe  $a$ . A probabilidade anterior da

classe  $a$  é denotada como  $p(a)$ , e a probabilidade anterior do preditor  $p$  é denotada como  $p(b)$ .

O *Naïve Bayes* é amplamente utilizado na classificação textos em várias classes e foi recentemente utilizado para classificação de análise de sentimentos.

### 3.0.4 Entropia máxima

O classificador Entropia máxima estima a distribuição condicional de uma classe marcada como um registro  $b$  utilizando um tipo de família exponencial com peso para cada restrição. O modelo com entropia máxima é o da família paramétrica  $P_{MaxEnt}(a|b)$  que maximiza a probabilidade.

Métodos numéricos, como escala iterativa e otimização quase-Newton, são geralmente empregadas para resolver o problema de otimização. O modelo é representado pelo seguinte:

$$P_{MaxEnt}(a|b) = \frac{\exp[\sum_i \alpha_i f_i(a, b)]}{\sum_a \exp[\sum_i \alpha_i f_i(a, b)]} \quad (12)$$

Onde  $a$  é a classe,  $b$  é o preditor e  $f_i$  é uma função binária que indica a presença de um atributo (característica) na entrada. O peso do vetor é indicado como  $\alpha_i$ .

### 3.0.5 Sentença e Level

Essa análise se concentra na classificação de sentenças em categorias, dependendo de serem positivas, negativas ou neutras. A análise de sentimentos do Twitter é considerada um exemplo de análise de sentimentos em nível de sentença.

O processo de análise de sentimentos do Twitter consiste em três etapas. Primeiro, o classificador é treinado usando conjuntos de dados que incluem tweets positivos, negativos e imparciais.

O segundo passo consiste na conversão de todos os textos do tweet para minúsculas. Além disso, os tweets devem ser limpos removendo URLs, caracteres hashtag ou menções de usuário, pois os métodos de análise de sentimentos do Twitter não se preocupam com esses caracteres. A etapa de pré-processamento inclui a filtragem de palavras de parada que são consideradas características discriminantes incomuns.

Após o pré-processamento, as previsões são realizadas. Nesta fase, vários algoritmos de

previsão, como SVM, classificador bayesiano e classificador de entropia, podem ser usados para decidir a polaridade de sentimentos dos tweets.

### 3.0.6 Trabalhos Relacionados à análise de sentimentos

Sharma et al. [24] propuseram um sistema de análise de sentimentos não supervisionado, baseado em documentos, capaz de determinar a orientação de sentimentos de documentos de texto com base em suas polaridades. A estratégia baseada em dicionário não supervisionada é utilizada como parte deste sistema, que também cuida da negação. Foi utilizado o WordNet que é um léxico adotado para definir vocabulários de opinião, suas palavras equivalentes e antônimos [24]. Neste estudo em particular, foram coletadas críticas de filmes para serem usadas como entrada, a fim de detectar o sentimento de polaridade dos documentos.

O sistema classificou cada um deles como positivo, negativo e imparcial e gerou resultados resumidos, apresentando o número total de documentos positivos, negativos e apartidários. Assim, o relatório resumido produzido pelo sistema ajudou os tomadores de decisão. Com esse sistema, a polaridade de sentimentos de qualquer documento é decidida com base na maioria dos vocabulários de opinião que aparecem nos documentos.

Chunxu Wu [27] propôs um método para sintetizar as orientações semânticas de opiniões dependentes do contexto que não podem ser determinadas usando o WordNet. O método proposto foi utilizado para decidir o sentimento de opinião, sendo utilizadas medidas de proximidade semântica. Essa abordagem baseou-se em tais medidas para determinar a orientação das revisões quando houve informações relevantes insuficientes. O experimento conduzido por Chunxu Wu [27] demonstrou que o procedimento proposto foi extremamente eficaz.

Harb et al. [28] propuseram uma abordagem de extração de sentimentos em nível de documento, concentrando-se em três estágios. Na primeira etapa, um conjunto de documentos que contém opiniões extraídas automaticamente da Internet. Em segundo lugar, os conjuntos de adjetivos positivos e negativos foram extraídos desse conjunto de dados de aprendizado. No terceiro estágio, novos conjuntos de testes de documentos foram classificados com base nas listas de adjetivos coletados no segundo estágio. Inúmeros experimentos foram conduzidos com dados reais e a abordagem proposta pelos autores alcançaram uma pontuação na F1 de 0,717 para identificar documentos positivos e uma pontuação na F1 de 0,622 para

reconhecer registros negativos.

Zagibalov et al. [29] abordaram a questão da classificação de sentimentos das revisões sobre produtos escritos em chinês. Sua abordagem se baseava em classificação não supervisionada capaz de ensinar a si mesma, aumentando a semente do vocabulário. Inicialmente, incluiu uma única palavra (boa) que foi marcada como positiva. A semente inicial foi treinada iterativamente para classificação de sentimentos. O critério de densidade de opinião foi então utilizado para calcular a proporção de sentimentos de um documento. Os experimentos mostraram que o classificador treinado atingiu um escore de 87% para detecção de polaridade de sentimentos após 20 iterações.

Já Tripathy e Rath [30] tentaram classificar revisões de acordo com sua polaridade usando algoritmos de aprendizado supervisionado, como o Naïve Bayes, o SVM, a Random Florest e a análise discriminante. Para conseguir isso, a abordagem proposta incluiu quatro etapas. Primeiro, a etapa de pré-processamento foi realizada para remover palavras de parada, caracteres numéricos e especiais. Segundo, as revisões de texto foram convertidas em uma matriz numérica. Terceiro, os vetores gerados foram utilizados como entradas para quatro classificadores diferentes. Os resultados foram obtidos posteriormente pela classificação de dois conjuntos de dados. Depois disso, várias métricas, como precisão, recall, f-measure e precisão de classificação, foram computadas para avaliar o desempenho da abordagem proposta. Para os conjuntos de dados de polaridade e IMDb, o classificador aleatório da floresta superou outros classificadores.

Saleh et al. [31] aplicaram o SVM a três conjuntos de dados diferentes para classificar as revisões de documentos. Vários esquemas de n gramas foram empregados para avaliar o impacto do SVM na classificação de documentos. Os pesquisadores utilizaram três abordagens de ponderação para gerar vetores de características a saber: Frequência de Termo Inverso de Frequência de Documento (TFIDF), Ocorrência Binária (BO) e Ocorrência de Termo (TO). Várias experiências foram então conduzidas para medir as combinações possíveis de várias abordagens de n-gramas e ponderação. Para o conjunto de dados Taboada, o melhor resultado de precisão foi obtido usando uma combinação do SVM com o TFIDF e o trigram. Para o corpus de Pang, os melhores resultados foram obtidos usando o BO e trigram. No que se refere ao corpus do SINAI, [?] mostraram que o classificador SVM alcançou o maior escore de precisão quando combinado com o TFIDF e o bigram.

Malhar e Ram [32] propuseram o método supervisionado para categorizar os dados do Twitter. Os resultados deste experimento demonstraram que o SVM teve um desempenho melhor que outros classificadores e, usando uma seleção de recurso híbrido, alcançou uma precisão de 88%. O experimento tentou combinar a análise de componentes principais (PCA) junto com o classificador SVM para reduzir a dimensionalidade dos recursos. Além disso, métodos de extração de recursos de unigrama, bigram, híbrido (unigram e bigram) foram utilizados. Esses autores demonstraram que a integração do PCA com o SVM com uma seleção de recursos híbridos poderia ajudar na redução das dimensões dos recursos e os resultados obtiveram uma precisão de classificação de 92%.

Anton e Andrey [33] desenvolveram um modelo para extrair polaridade de sentimentos dos dados do Twitter. As características extraídas foram palavras contendo n-gramas e emoticons. O experimento realizado demonstrou que o SVM teve um desempenho melhor que o Naïve Bayes. O melhor método de desempenho geral foi o SVM em combinação com a extração de recursos de unigrama, alcançando uma precisão de 81% e uma precisão de recall de 74%.

Po-Wei Liang et al. [34] projetaram uma estrutura chamada “minerador de opinião” que investigou e detectou automaticamente os sentimentos das mensagens de mídia social. Os tweets analisados foram combinados para a realização da análise e, nesse contexto, foram extraídas mensagens que continham sentimentos (os tweets sem opinião foram removidos) e suas polaridades determinadas (positivas ou negativas). Para conseguir isso, os pesquisadores [34] classificaram os tweets em “opinião” e “Não opinião” usando o classificador NB com um unigrama. Eles também descartaram recursos irrelevantes utilizando a estratégia de Informações Mútuas e extração de qui-quadrado. Os resultados experimentais confirmaram a adequação da estrutura para análise de sentimentos em aplicativos de microblog genuínos.

Kouloumpis et al. [35] exploraram a utilidade de vários recursos linguísticos para extrair os sentimentos dos dados do Twitter. Os conjuntos de dados com hash (HASH) e emoticons (EMOT) foram utilizados para treinar os classificadores e o conjunto de dados iSieve foi usado para a avaliação. Neste estudo, vários conjuntos de recursos foram introduzidos usando unigramas, bigrams, léxicos, micro-blogs e elementos de parte do discurso. O classificador AdaBoost foi treinado usando esses recursos selecionados em diferentes combinações. Os resultados mostraram que os recursos de partem do discurso eram ruins para a

análise de sentimentos dos dados do Twitter, enquanto os recursos de microblogging eram os mais úteis. Os melhores resultados foram alcançados quando os recursos de n-grama foram empregados juntamente com os recursos de léxico e microblogging. Um escore F de 0,68 foi obtido com os conjuntos de dados HASH e um escore F de 0,65 com os conjuntos de dados HASH e EMOT combinados.

Hamdan et al. [36] extraíram diferentes tipos de características com a intenção de aumentar a precisão da classificação de sentimentos. Os recursos do Unigram foram introduzidos como linha de base, enquanto as palavras foram consideradas recursos independentes. Recursos específicos de domínios também foram incluídos, como o número de retweets. O DBpedia foi utilizado para explorar os conceitos contidos nos tweets; estes foram denominados recursos DBpedia. O WordNet foi usado para identificar os sinônimos de substantivos, verbos, advérbios e adjetivos. O SentiWordNet foi empregado para calcular a frequência de palavras positivas e negativas que aparecem nos tweets e as polaridades deles. As experiências mostraram que a adição de adjetivos, recursos SentiWordNet e DBpedia, levou a pequenas melhorias na precisão do SVM e do NB. As proporções dessas pequenas melhorias foram de aproximadamente 2% com o SVM e 4% com o RN.

## 4 Metodologia

Esta seção aborda as etapas metodológicas utilizadas para este trabalho, descrevendo cada uma das etapas que formam a ferramenta proposta. Merece destaque que foi utilizado a linguagem de programação Python, por possuir uma vasta coleção de ferramentas pre-definidas que auxiliaram no desenvolvimento do trabalho.

### 4.1 Extração de dados

#### 4.1.1 *Twitter*

A coleta dos dados foi realizada na rede social Twitter, utilizando as palavras chaves “Petrobras”, “PETR4”, “Banco do Brasil” e “BBAS3”. Para a extração dos dados, foi analisado os tweets relacionado a essas palavras. Essa coleta pode ser realizada de forma manual, onde o usuário após fazer a busca pelos termos, copia e lê os tweets relacionados, verificando um a um e eliminando aqueles que destoam da busca ou são repetidos. Entretanto, desta forma, o trabalho e tempo despendido para a análise estaria aquém do esperado e não seria totalmente eficaz.

Como forma de contornar a situação, é possível utilizar a API do Twitter baseada em *Python* como forma de filtragem dos resultados. Neste caso, através da adição de uma rotina, na qual ele cataloga os tweets relacionados às palavras chaves antes definidas, e os lista, possibilitando a sua coleta e posterior análise, não necessitando diretamente de trabalho manual. Imperioso abordar que existem duas APIs nessa rede social, onde uma é paga e a outra gratuita, entretanto essa segunda possui limitação de até sete dias anteriores a data da coleta. Sendo assim, as coletas aqui realizadas foram feitas todos os domingos às 18hs, abrangendo o período de primeiro de julho de 2019 até 15 de abril de 2020.

Os tweets não foram coletados de modo geral da rede social, primando por aqueles escritos e publicados pelos perfis "bbc brasil"; "folha mercado"; "g1 economia"; "olhar digital"; "UOL Economia"; "valor economico" e "O Globo Economia" por serem de meios de comunicação da imprensa e confiáveis no meio econômico. Entretanto existem outros daytraders disponíveis na rede que podem auxiliar a tomada de decisão, porém utilizou-se esses por serem de fácil acesso.



### 4.1.2 Yahoo Finances

Utilizando uma API do Yahoo Finance baseada em Python, foi extraída duas series históricas da Petrobras e do Banco do Brasil. Para tanto, utilizou-se na plataforma desta API as palavras chaves “PETR4.SA” e “ITUB4.SA”. A primeira série partiu do dia 01/01/2010 à 15/04/2020 e a segunda série do dia 01/07/2019 a 15/04/2020 para ambas as empresas. Diferente da API do Twitter, a do Yahoo Finance é livre, não possuindo limitação de série histórica, possibilitando maior coleta de dados. Os dados extraídos tiveram como fonte de origem a Bolsa Brasileira de Valores (BOVESPA).

## 4.2 Extração de dados

### 4.2.1 Tratamento de dados do Twitter

Após a exportação dos tweets para o Excel, foi realizado manualmente a exclusão de spams e tweets duplicados. A exclusão manual desses tweets era necessária para garantir uma acurácia precisa, pois na fase de classificação, tweets duplicados poderiam ser classificados de modo divergentes, o que poderia prejudicar a acurácia da análise. Além disto, foi empregado uma ferramenta denominada *Natural Language Toolkit* (NLTK) que está presente no *Python*, que age como um tokenizador, ou seja, ele “quebra” as frases do *tweet* em palavras isoladas, facilitando a sua classificação.

Também foi feita nessa fase a remoção de stop words, que são palavras utilizadas em demasia na língua e que não afeta a classificação do texto, o que facilita a sua classificação ao suprimi-las. Na língua portuguesa, essas stop words podem ser classificadas, de modo geral como os artigos definidos e as preposições.

### 4.2.2 Tratamento de dados do Twitter

Após extração dos dados, foram deletados manualmente os campos vazios pois eles não possuíam valores de análise, em seguida foi realizado um *dataFrame* para que fosse possível organizar os dados que eram necessários na análise, sendo eles: Abertura (*Open*), Fechamento (*Close*) e Volume de Negociação (Volume). A partir desse ponto, com os dados de treino, realizou-se uma normalização dos dados, que consiste na redução da escala dos valores para um conjunto de escala comum, ou seja, padronizar os valores de cada variável para

que a comparação entre elas seja o mais justo possível, além de evitar *overfitting*, também conhecida como a maldição da dimensionalidade, por ocorrer o hiper parâmetro de conjunto de valores.

### 4.3 Classificador

Para submeter os dados ao software foi necessário realizar uma classificação manual onde os tweets foram submetidos ao crivo de um leigo sobre finanças, para que os classificasse em positivo ou negativo, baseado em seus conhecimentos próprios.

Após isso, os dados do Twitter foram tratados utilizando os classificadores baseados no Naive Bayes e SVM. O primeiro classificador, se utiliza de frequências, onde primeiramente ele cria uma base de dados com todas as palavras do dataset, e atribui um valor a cada uma, onde 1 é positivo e 0 é negativo. A partir daí ele substitui cada palavra da frase por um valor e a sua soma é dividido pela frequência da sentença.

Para fins de comparação, os dados foram classificação, novamente no SVM que funciona através da separação das classes “positivo” e “negativo” em um hiperplano, traçando limites entre eles, classificando-os e ao mesmo tempo aperfeiçoando sua análise.

Essa base foi dividida em 70% como treinamento e 30% como teste, onde as medidas para quantificar os acertos feitos pelo classificador foram acurácia, precisão, *recall* e *F-Measure*.

#### 4.3.1 Acúracia

É definida como a quantidade de acertos realizados pelo classificador como um todo, sendo calculada pela equação (13):

$$Acuracia = \frac{VP + VN}{VP + FP + VN + FN} \quad (13)$$

onde:

- VP - Verdadeiro Positivo
- VN - Verdadeiro Negativo
- FP - Falso Positivo
- FN - Falso Negativo

### 4.3.2 Precisão

A precisão é o resultado da quantidade de assertivas verdadeiramente positivas/negativa dentre aquelas que foram classificadas como positivas/negativas, podendo ser expressa pela equação (14):

$$Precisao = \frac{VP}{VP + FP} \quad (14)$$

### 4.3.3 Recall

A equação (15) expressa a quantidade de assertivas positivas/negativas que deveriam ter sido classificadas como positivo/negativo no universo da amostra.

$$Recall = \frac{VP}{VP + FN} \quad (15)$$

### 4.3.4 F-Measure

Também é chamada na literatura por F1-score ou apenas F-Score. Se trata da media ponderada entre as medidas de precisão e de recall quando tratamos de combinações binárias, sendo expostas pela equação (16):

$$F - Measure = \frac{2 \cdot Precisao \cdot Recall}{Precisao + Recall} \quad (16)$$

## 4.4 Redes LSTM

Considerando o volume de notícias/tweets que são produzidos por dia, foi necessário realizar uma análise para determinar as tendências das notícias dos dias que apresentaram mais de uma notícia, classificando-a em positiva ou negativa. Isto se deve por os dados da série histórica serem diários.

Após a classificação dos tweets, os dados foram importados para a rede neural LSTM. Neste ambiente, os dados do Twitter foram incorporados a uma série histórica de dez meses (entre 01/07/2019 a 15/04/2020) dos dados da API do Yahoo Finance com o intuito de verificar se a análise de sentimentos das notícias do Twitter ajudam a predizer um ativo.

Paralelamente a instancia dos dados do *Twitter* e do *Yahoo Finance*, foi submetido os dados da série histórica de dez anos do Yahoo Finance. Nesta segunda instancia, não houve adição dos dados do Twitter por não haver dados que abrangesse a totalidade do dataset. O intuito da rede foi de prever os ativos para os próximos dez dias.

#### 4.5 *Deep FeedForward Neural Network (DNN)*

A fim de comparativo os dados foram também analisados em uma rede neural profunda, a DNN. Isto se deu, porque essa rede neural possui mais camadas e neurônios que podem melhorar os algoritmos de predição frente àqueles mais tradicionais como a LSTM. Assim como a anterior, esta rede teve como objetivo prever os ativos para os próximos dez dias.

#### 4.6 Validação

A validação para as redes foram feitas pela métrica *Root Mean Squared Error (RMSE)*, que calcula a raiz quadrática média dos erros entre os valores observados (reais) e predições (hipótese) dado pela fórmula (17):

$$rmse = \sqrt{\left| \frac{1}{n} \right| \sum_{i=1}^n (y_i - x_i)^2} \quad (17)$$

Onde:

- $y_i$  - Valor previsto;
- $x_i$  - Valor verdadeiro;
- $n$  - Número total de dados

## 5 Resultados e Discussão

Neste capítulo procurou-se descrever os resultados do processamento e treinamento dos dados nas redes neurais. Na seção atual será detalhado como o classificador se comporta com a introdução de novos tweets, em seguida os resultados que foram obtidos em cada rede neural, sem e com a análise de sentimentos.

Primeiramente será apresentado os resultados com relação ao classificador, mostrando como ele se comportou com a introdução de tweets sem pré-processamento de sentimento, apresentando os respectivos valores da acurácia, precisão, *recall* e *F-Measure*.

Em seguida são apresentados os resultados referentes à predição realizada nas redes neurais sem a análise de sentimentos, com o intuito de comparar e identificar pontos em comum ou não, com a predição com análise de sentimentos.

No tópico seguinte são apresentados os resultados obtidos com a análise de sentimentos, servindo de substrato para o último tópico que versa sobre os comparativos entre as duas formas de predição aqui adotada.

### 5.1 Resultado dos classificadores

A importância dos classificadores nessa etapa do processo é de grande relevância, por tornar o trabalho de predição mais autônomo e com menores intervenções humanas, diminuindo possíveis julgamentos de juízo do programador, o que pode tornar os resultados tendenciosos.

#### 5.1.1 Comportamento do *Naive Bayes*

Na tabela 2 está expressa a matriz de confusão na qual utilizando o vetorizador TF-IDF estão dispostas as análises de sentimentos referentes a empresa Petrobras S.A., no qual conteve 772 notícias/tweets, que perfazem o período de 01/07/2019 a 15/04/2020. O vetorizador TF-IDF atribui valores para as palavras, as que são mais frequentes é atribuído um valor baixo, e as menos frequentes um valor mais alto. O algoritmo entende que as palavras mais repetidas não são tão importantes, as menos repetidas são mais relevantes ou importantes na classificação entre positivo e negativo. Os resultados plotados demonstram que ainda que apresentem limitações, as análises foram abaixo do esperado.

Ainda na tabela 2 pode ser observado a matriz de confusão para o Banco do Brasil S.A., pautado nas análises de sentimento, utilizando a mesma metodologia para a matriz de confusão da Petrobras, tendo o mesmo período de coleta de dados, contudo apresentando 150 notícias/tweets. Seguindo a mesma tendência, a classificação desses dados foram, também ruins.

Tabela 2: Matriz de confusão do classificador de Naive Bayes para as empresas Petrobras e Banco do Brasil

	Petrobras		Banco do Brasil	
	Negativo	Positivo	Negativo	Positivo
Negativo	51	347	03	04
Positivo	28	106	06	28

Também considerou-se trabalhar com os dados da empresa Vale S.A. porém, devido o nome da empresa ser um verbo comum na língua portuguesa, ao fazer a filtragem dos tweets, a quantidade de informação era extensa e em sua maior parte não condizia com a empresa.

### 5.1.2 Comportamento do SVM

Os mesmos dados foram incluídos no classificador SVM, o qual apresentou a matriz de confusão contida na tabela 3. Diferentes dos resultados do classificador *Naive Bayes*, o SVM apresentou números mais satisfatórios.

Um comportamento relevante do SVM em comparação ao Naive Bayes foi que ele conseguiu classificar algumas notícias/tweets verdadeiras-negativas, apesar de ainda assim ter apresentado falsos-negativos mais altos. Isto se deve à baixa quantidade de notícias/tweets disponíveis no dataset quando comparadas com as da Petrobras e, por ter poucas notícias/tweets negativas na classificação manual.

Tabela 3: Matriz de confusão do classificador de SVM para Petrobras e Banco do Brasil

	Petrobras		Banco do Brasil	
	Negativo	Positivo	Negativo	Positivo
Negativo	63	35	02	05
Positivo	24	110	05	29

O desempenho desse classificador quando comparado ao Naive Bayes (Tabela 2), de modo geral foi superior, podendo ser considerada como uma boa análise, apresentando acurácia de 74,56% para os dados da Petrobras, contudo vale destacar de que mesmo com esses resultados, o SVM apresenta margens para melhorar sua acurácia.

Referente aos dados do classificador SVM para o Banco do Brasil, apresentou acurácia de 75,60%, de modo geral, foi um resultado identico quando comparado com o Naive Bayes.

Os resultados do classificador Naive Bayes, referentes a adjetivação das notícias/tweets para a Petrobras foram considerados regulares, uma vez que apresentaram acurácia de 67,67%. Sobre os dados do Banco do Brasil, guardada as explicações expostas anteriormente, houve acurácia de 75.61%. A fim de detalhamento, na tabela 4 estão plotadas as medidas de quantificação das análises.

O intuito de utilizar dois classificadores foi comparar e escolher o melhor classificador que apresentasse o melhor desempenho. Sendo assim, constata-se que o classificador SVM apresenta melhores cenários na classificação de notícias/tweets que auxiliam na predição de ativos.

Tabela 4: Comparativo das medidas de quantificação dos classificadores Naive Bayes e SVM

	<b>Petrobras</b>		<b>Banco do Brasil</b>	
	<b>Naive Bayes</b>	<b>SVM</b>	<b>Naive Bayes</b>	<b>SVM</b>
<b>Acurácia</b>	0.6767	0.7456	0.7561	0.7560
<b>Precisão</b>	0.7910	0.8208	0.8235	0.8529
<b>Recall</b>	0.6928	0.8750	0.8292	0.8529
<b>F-Measure</b>	0.7386	0.8484	0.9066	0.8529

## 5.2 Predição sem análise de sentimentos

Neste momento, começou-se a realizar as análises das series históricas nas redes neurais LSTM e DNN. Abaixo estão as análises, onde foi comparado diferentes configurações de redes, com o intuito de melhor otimizar as previsões e encontrar aquela que apresentou as predições mais próximas com as reais. Essas predições foram realizadas com os dados da série histórica de dez anos do Yahoo Fincance, nas ações da Petrobras e Banco do Brasil.

### 5.2.1 Rede DNN

Na Figura 9 foi realizada a predição dos preços da Petrobras (Figura 9.A) e do Banco do Brasil. Os hiper parâmetros definidos para cada ação estão dispostos na tabela 5.

Tabela 5: Hiper parâmetros para as ações da Petrobras e Banco do Brasil da rede DNN

	BANCO DO BRASIL	PETROBRAS
DNN	7 CAMADAS (5 OCULTA)	5 CAMADAS (3 OCULTA)
	Dropout = 0	Dropout = 0
	Batch_size = 100	Batch-Size = 100
	Epochs = 100	Epochs = 50
	Validação = 0.05	Validação = 0.05
	Função de Ativação = ReLU	Função de ativação = ReLU

Ao fim, a previsão da Petrobras apresentou *Root mean squared error* (RMSE) de 0.1389 e a do Banco do Brasil RMSE de 0.1097 sendo, esta segunda a melhor predição.

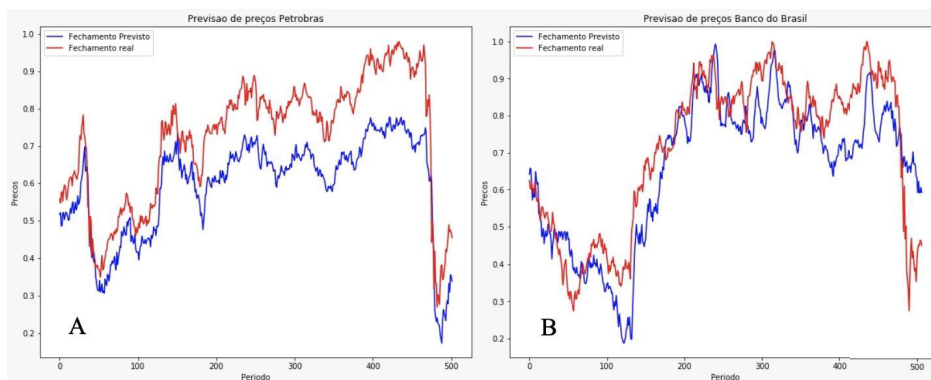


Figura 9: Gráficos das previsões de ativos para a Petrobras (A) e para o Banco do Brasil (B) na rede DNN para o período de 10 anos. Fonte: Yahoo Finance (2020), copilado pelo autor

### 5.2.2 Rede LSTM

Os mesmos dados do Yahoo Finance para a Petrobras e Banco do Brasil também foram inseridos na rede LSTM (Figura 10). Os hiper parâmetros definidos para cada ação estão dispostos na tabela 6.



Tabela 6: Hiper parametros para as ações da Petrobras e Banco do Brasil da rede LSTM

	BANCO DO BRASIL	PETROBRAS
LSTM	3 CAMADAS (1 OCULTA)	3 CAMADAS (1 OCULTA)
	Dropout = 0.2	Dropout = 0.2
	Batch_size = 128	Batch-Size = 100
	Epochs = 100	Epochs = 50
	Validação = 0.05	Validação = 0.05
	Função de Ativação = ReLU	Função de ativação = Linear

Com relação ao RMSE, enquanto que a predição da Petrobras foi de 0.1011, a do Banco do Brasil ficou em 0.0716. Neste cenário, assim como na rede DNN, o Banco do Brasil foi a empresa que demonstrou a melhor predição. Isto reflete diretamente pela maior estabilidade dos dados do Banco do Brasil comparado com os dados da Petrobras.

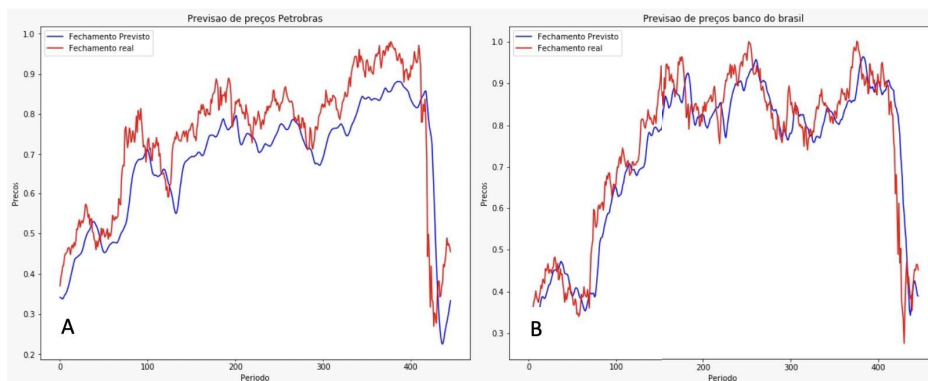


Figura 10: Gráficos das previsões de ativos para a Petrobras (A) e para o Banco do Brasil (B) na rede LSTM para o período de 10 anos. Fonte: Yahoo Finance (2020), copilado pelo autor

### 5.3 Predição com análise de sentimentos

Para a análise de sentimentos, foram adicionados 196 dados, extraídos das notícias/tweets, que corresponderam às tendências citadas na metodologia, abrangendo o período de 01/07/2019 a 15/04/2020. A ideia central de realizar tal ato, foi para saber se existe alguma ligação direta, ou indireta, dos preços das ações com as notícias que são vinculadas na mídia.

Assim como os dados da série histórica sem análise de sentimentos, essas também foram submetidas tanto à rede DNN como LSTM, com o intuito de encontrar as principais diferenças entre elas e precisar qual a mais eficaz. Como já citado anteriormente, nessa fase da análise das previsões, elas ocorreram apenas com os dados da Petrobras.

Na Figura 11.A está plotado a predição dos preços para a Petrobras em conjunto com a análise de sentimentos, onde foi utilizada a rede LSTM e na Figura 11.B estão os dados da predição para o Banco do Brasil na rede LSTM. Os hiper parâmetros definidos para cada ação estão dispostos na tabela 7

Tabela 7: Hiper parâmetros para as ações da Petrobras e Banco do Brasil da rede LSTM com análise de sentimento

	BANCO DO BRASIL	PETROBRAS
LSTM	3 CAMADAS (1 OCULTA)	3 CAMADAS (1 OCULTA)
	Dropout = 0.2	Dropout = 0.2
	Batch_size = 128	Batch-Size = 128
	Epochs = 100	Epochs = 50
	Validação = 0.05	Validação = 0.05
	Função de Ativação = ReLU	Função de ativação = Linear

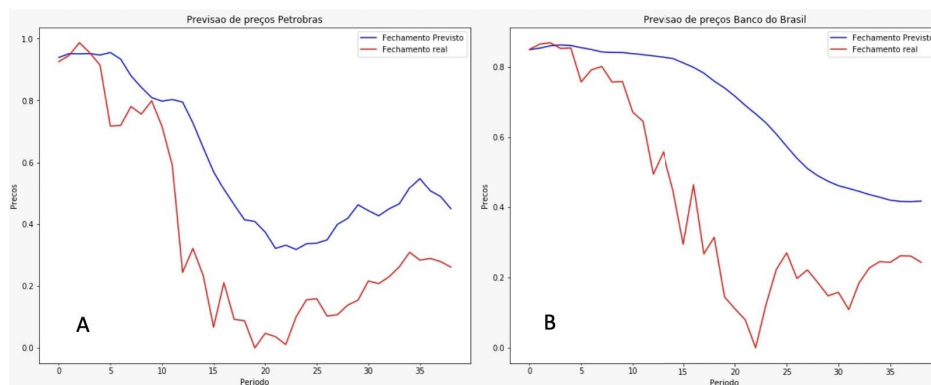


Figura 11: Gráficos das previsões de ativos para a Petrobras (A) e do Banco do Brasil (B) na rede LSTM com adição da análise de sentimento para o período de 9 meses. Fonte: Yahoo Finance (2020), copilado pelo autor.

Por sua vez, na Figura 12.A os mesmos dados da Petrobras foram inseridos em uma rede

DNN. E na Figura 12.B, os mesmos dados do Banco do Brasil foram inseridos também em uma rede DNN. Os hiper parâmetros definidos para cada ação estão dispostos na tabela 8.

Tabela 8: Hiper parâmetros para as ações da Petrobras e Banco do Brasil da rede DNN com análise de sentimento

	BANCO DO BRASIL	PETROBRAS
DNN	6 CAMADAS (4 OCULTA)	5 CAMADAS (3 OCULTA)
	Dropout = 0	Dropout = 0
	Batch_size = 50	Batch-Size = 100
	Epochs = 100	Epochs = 50
	Validação = 0.01	Validação = 0.01
	Função de Ativação = ReLU	Função de ativação = ReLU

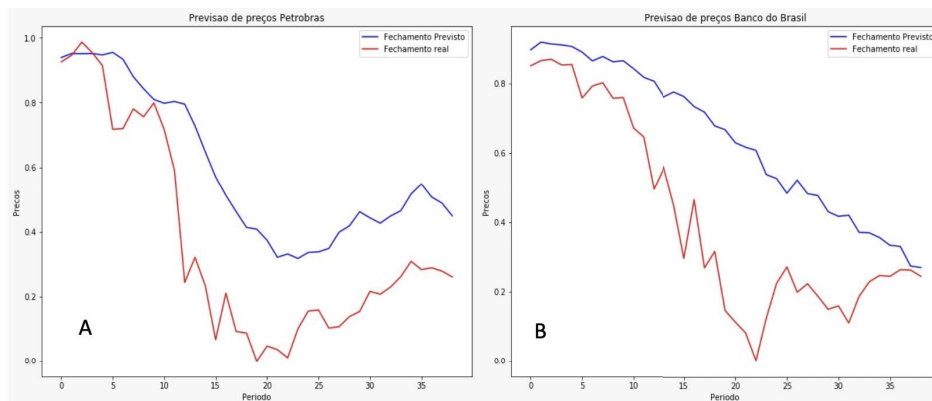


Figura 12: Gráficos das previsões de ativos para a Petrobras (A) e do Banco do Brasil (B) na rede DNN com adição da análise de sentimento para o período de 9 meses. Fonte: Yahoo Finance (2020), copilado pelo autor.

Com relação ao RMSE das quatro previsões (Tabela 9) as DNN's tiveram desempenho melhor do que às LSTM's, para ambas as empresas.

Tabela 9: Comparativo dos RMSE das redes neurais LSTM e DNN

	Petrobras	Banco do Brasil
<b>LSTM</b>	0.3513	0.3270
<b>DNN</b>	0.2641	0.2425

As redes LSTM são mais usadas para os métodos de predição e se mostraram bastante eficazes, por sua vez, as redes DNN mostraram resultados satisfatórios, isto pode ser observado quando ao analisar os resultados de predição do Banco do Brasil sem análise de sentimentos a rede DNN exibiu resultados levemente inferiores, apesar de não apresentarem valores tão distantes da rede LSTM, evidenciando que as redes DNN podem realizar predições tão eficazes quanto a LSTM.

As redes LSTM por possuírem uma memória residual, conseguem buscar em seu acervo informações que auxiliam na análise dos dados, podendo assim prever de modo mais satisfatório. As redes DNN por conseguinte, por utilizarem mais camadas ocultas possibilita maior capacidade de aprendizado, fazendo com que a rede reconheça com mais facilidade os padrões numéricos mesmo em dados não estruturados.

## **5.4 Discussão**

Como pode ser observado, os resultados apresentados nas sessões anteriores, ainda que não tenham sido os ideais para prever com maior fidedignidade os ativos da Petrobrás e Banco do Brasil na bolsa de valores brasileira, apresentaram resultados satisfatórios.

Para a Petrobrás foram extraídos no total 835 notícias/tweets de jornais econômicos bem conceituados no Brasil. Esses dados passaram por um processo de pré-tratamento onde foram eliminadas notícias duplicadas e notícias em que não teriam influência na bolsa de valores, priorizando notícias políticas, ações sociais, alta e baixa de lucros, previsões a cada trimestre, e notícias que possam melhorar ou manchar a credibilidade da empresa frente a sociedade e investidores.

O mesmo foi feito para a empresa do Banco do Brasil, porém a quantidade de notícias/tweets foram bem menores, tendo um total de 160 notícias/tweets, reduzindo para 150 depois de feito o tratamento dos dados. A quantidade menor de notícias comparada aos dados da bolsa que foram 196 dados diários levou-se a metodologia de repetir para o dia em que não houve nenhuma notícia. Para mais de uma notícia em um mesmo dia, realizou-se uma contagem, sendo a tendência positiva ou negativa do dia levado em consideração a quantidade de notícias positivas ou negativas naquele dia. A limitação dos dados a duas empresas e a escassez de notícias referentes ao Banco do Brasil, ainda que possam prejudicar na análise mais detalhada, não inviabiliza a sua predição. A extração de dados foi feita utilizando a API

do Twitter, que permite que se extraia 100 tweets em um intervalo de sete dias, indicando qual palavra-chave procurar em seu corpo e limitando quais perfis a API deve rastrear. Tudo isso foi feito numa linguagem em Python, utilizando o Notebook Jupyter.

Os classificadores escolhidos, SVM e Naive Bayes, também foram feitos no mesmo programa. Os resultados apresentados foram satisfatórios, com o Naive Bayes apresentando desempenho menor que o SVM nos dados da Petrobras, o que está de acordo com a literatura, mas apresentando uma acurácia tecnicamente idêntica na empresa Banco do Brasil. Porém é bom ressaltar que na empresa do Banco do Brasil a quantidade de dados disponíveis foram muito baixas, impactando diretamente no desempenho dos classificadores. Os resultados desse trabalho foram próximos aos encontrados na literatura, como mostra os trabalhos de Cakra e Trisedya [20] e Anton e Andrey [33]

Como pesquisa futura, pode-se fazer uma combinação híbrida para melhorar sua acurácia, bem como coletar uma quantidade de dados maior, e não apenas em um intervalo de 9 meses, como mostra a pesquisa de Malhar e Ram [32].

A importância de se utilizar mais de um classificador para comparação é possibilitar a visualização de dois classificadores diferentes executando a mesma atividade e assim, poder apurar aquele que apresenta o melhor desempenho.

A rede neural artificial escolhida para esse trabalho, foi a LSTM, por ser a mais utilizada neste tipo de previsão, pois elas possuem uma memória residual e conseguem buscar em seu acervo informações que auxiliam na análise de dados.

As redes DNN são relativamente novas, tendo boa aplicabilidade em reconhecimento de padrões de figuras, sendo muito utilizadas por gigantes da indústria como a Google, Apple e Nvidia. Na área de previsões de ativos elas não costumam ser muito utilizadas, principalmente por não conterem em seu acervo as memórias residuais que contêm nas redes LSTM. Porém, por utilizarem uma maior quantidade de camadas ocultas, ela aumenta sua capacidade de aprendizado, fazendo com que reconheça com mais facilidade padrões numéricos em dados não estruturados.

A divisão das análises das redes contendo a análise de sentimento se deu pelo motivo de comparar e entender até onde as notícias divulgadas na mídia podem ou não afetar a alta ou baixa de uma ação. Uma vez que já foi observado que há sim influência das notícias com os preços das ações no trabalho de Cakra e Trisedya [20].

Nota-se que as redes LSTM e DNN apresentaram bons resultados, entretanto a LSTM apresentou pequenas margens de diferença quando comparadas às redes DNN, o que mostra que as redes DNN podem ser uma boa alternativa para realizar predições.

Quando acrescentamos ao modelo as análises de sentimentos oriundas dos classificadores, as redes perderam desempenho quando comparada as mesmas redes sem análise de sentimento. Paradoxalmente ao que foi percebido anteriormente, neste novo cenário, as redes DNN apresentaram melhores resultados comparadas às redes LSTM. Isso pode ter ocorrido devido a maior quantidade de camadas ocultas das redes DNN, que conseguiram reconhecer melhor a relação entre notícias e os preços de fechamento.

É relevante lembrar que a quantidade baixa de entrada (196 dias) afetaram negativamente o desempenho das duas redes, pois precisam de uma quantidade de dados maior para oferecer melhor desempenho. Outro fato que deve ser considerado é o período de crise enfrentado pelo Brasil e o mundo, por conta da pandemia do Coronavírus, enfatizando que as redes neurais têm dificuldades na predição em períodos de crises excepcionais.

Como trabalhos futuros, uma quantidade maior de dados extraídos de outros portais, ou até mesmo de usuários comuns para ajudar a melhorar os classificadores e consequentemente alimentar com mais qualidade as redes neurais, bem como utilizar Deep recurrent neurais network para verificar seu desempenho frente às outras redes já citadas.

## 6 Conclusão

Neste trabalho foi proposto que se realizasse a predição de ativos através da análise de sentimentos em redes neurais. Assim exposto, pode-se afirmar que o objetivo geral deste trabalho foi alcançado, sendo desenvolvido com sucesso um modelo que fosse capaz de prever os ativos baseado em notícias/tweets.

Os resultados dos classificadores ainda que sejam parecidos, demonstrou que o SVM conseguiu melhor desempenho. Porém, devido à baixa quantidade de notícias/tweets, fez com que o classificador não conseguisse ter em seu dataset um acervo suficientemente grande que lhe proporcionasse uma análise mais aprofundada. Além disto, considera-se que essa limitação na quantidade de notícias/tweets classificados e que foram introduzidos na rede possam ter impactado diretamente nas predições ficando com acurácia girando entre 67,67% e 74,56%.

Com relação as redes neurais, as DNN demonstraram bom poder preditivo, fato este que deve ser considerado, uma vez que em sua maioria, as predições são feitas utilizando redes LSTM, mostrando que redes *deep recurrents neural network* podem apresentar desempenho ainda melhor, em tese.

A análise de sentimentos em si, nas redes neurais ficaram abaixo do esperado, tendo em vista que em outras pesquisas há relação entre o sentimento das notícias com o preço mesmo que não sejam variações tão expressivas. O baixo número de dados impactou diretamente no aprendizado da rede, ocasionando valores de RMSE elevados.

Além disto, o arranjo social e econômico durante o período de análise dos dados não favoreciam as predições por conta da pandemia do Corona Vírus, sendo, portanto, uma área fecunda para maiores investigações, os dados mesmo sendo promissores abrem espaço para mais pesquisas que possam aperfeiçoar os mecanismos de predição, reduzindo os erros e melhorando a precisão.

Em trabalhos futuros pode-se trabalhar com híbridos de SVM visando uma melhor categorização dos dados, além de testar com diferentes tipos de notícias e de outras empresas/multinacionais. Fica-se o desafio de tentar utilizar uma rede *deep recurrent neural network*, testando-os com diferentes bases de dados e bolsas de valores, primando para uma melhor otimização das redes aqui propostas.

## Referências

- [1] Katti Faceli, Ana Carolina Lorena, João Gama, André Carlos Ponce de Leon Carvalho, et al. *Inteligência artificial: Uma abordagem de aprendizado de máquina*. 2011.
- [2] Apostolos Nicholas Refenes, Achileas Zaprakis, and Gavin Francis. Stock performance modeling using neural networks: a comparative study with regression models. *Neural networks*, 7(2):375–388, 1994.
- [3] Maria Lígia Chuerubim and Irineu da Silva. Possibilidades em inteligência artificial na detecção de padrões e previsão de acidentes em rodovias. 2019.
- [4] Solange Oliveira Rezende. *Sistemas inteligentes: fundamentos e aplicações*. Editora Manole Ltda, 2003.
- [5] Rodrigo MS Soares, Vitor H Ferreirab, and Márcio Z Fortesc. Rede neural aplicada em sistemas de proteção catódica—estudo de caso. 2018.
- [6] Luca Massaron John Mueller. *Machine Learning For Dummies*, volume 1 of 1. Wiley, 111 River Street, Hoboken, New Jersey, 3 edition, 12 2019. ..
- [7] Tommy Jung Anasse Bari, Mohamed Chaouchi. *Análise Preditiva para Leigos*, volume 1 of 1. Wiley, 111 River Street, Hoboken, New Jersey, 3 edition, 12 2019. ..
- [8] Thiago Bianchi. *MaSTA: a text-based machine learning approach for systems-of-systems in the big data context*. PhD thesis, Universidade de São Paulo, 2019.
- [9] Zsolt László Kovács. *Redes neurais artificiais*. Editora Livraria da Física, 2002.
- [10] Sandro Skansi. *Introduction to Deep Learning: from logical calculus to artificial intelligence*. Springer, 2018.
- [11] Hans L Bodlaender, John R Gilbert, Hjálmtyr Hafsteinsson, and Ton Kloks. Approximating treewidth, pathwidth, frontsize, and shortest elimination tree. *J. Algorithms*, 18(2):238–255, 1995.



- [12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [13] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. Survey of dropout methods for deep neural networks. *arXiv preprint arXiv:1904.13310*, 2019.
- [14] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116, 1998.
- [15] Masaya Abe and Hideki Nakayama. Deep learning for forecasting stock returns in the cross-section. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 273–284. Springer, 2018.
- [16] Wei Bao, Jun Yue, and Yulei Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one*, 12(7), 2017.
- [17] Vishal Kharde, Prof Sonawane, et al. Sentiment analysis of twitter data: a survey of techniques. *arXiv preprint arXiv:1601.06971*, 2016.
- [18] Tom Fawcett and Foster Provost. Activity monitoring: Noticing interesting changes in behavior. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 53–62, 1999.
- [19] Gabriel Pui Cheong Fung, Jeffrey Xu Yu, and Wai Lam. News sensitive stock trend prediction. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 481–493. Springer, 2002.
- [20] Yahya Eru Cakra and Bayu Distiawan Trisedya. Stock price prediction using linear regression based on sentiment analysis. In *2015 international conference on advanced computer science and information systems (ICACSIS)*, pages 147–154. IEEE, 2018.
- [21] Ann Devitt and Khurshid Ahmad. Sentiment polarity identification in financial news: A cohesion-based approach. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 984–991, 2007.

- [22] Ritika Singh and Shashi Srivastava. Stock prediction using deep learning. *Multimedia Tools and Applications*, 76(18):18569–18584, 2017.
- [23] Rui Xia, Chengqing Zong, and Shoushan Li. Ensemble of feature sets and classification algorithms for sentiment classification. *Information sciences*, 181(6):1138–1152, 2011.
- [24] Richa Sharma, Shweta Nigam, and Rekha Jain. Polarity detection at sentence level. *International journal of computer applications*, 86(11), 2014.
- [25] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.
- [26] Jayashri Khairnar and Mayura Kinikar. Machine learning algorithms for opinion mining and sentiment classification. *International Journal of Scientific and Research Publications*, 3(6):1–6, 2013.
- [27] Chunxu Wu, Lingfeng Shen, and Xuan Wang. A new method of using contextual information to infer the semantic orientations of context dependent opinions. In *2009 International Conference on Artificial Intelligence and Computational Intelligence*, volume 4, pages 274–278. IEEE, 2009.
- [28] Ali Harb, Michel Plantié, Gerard Dray, Mathieu Roche, François Trouset, and Pascal Poncelet. Web opinion mining: How to extract opinions from blogs? In *Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology*, pages 211–217, 2008.
- [29] Taras Zagibalov and John A Carroll. Unsupervised classification of sentiment and objectivity in chinese text. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*, 2008.
- [30] Abinash Tripathy and Santanu Kumar Rath. Classification of sentiment of reviews using supervised machine learning techniques. *International Journal of Rough Sets and Data Analysis (IJRSDA)*, 4(1):56–74, 2017.
- [31] M Rushdi Saleh, Maria Teresa Martín-Valdivia, Arturo Montejo-Ráez, and LA Ureña-López. Experiments with svm to classify opinions in different domains. *Expert Systems with Applications*, 38(12):14799–14804, 2011.

- 
- [32] Malhar Anjaria and Ram Mohana Reddy Guddeti. Influence factor based opinion mining of twitter data using supervised learning. In *2014 Sixth International Conference on Communication Systems and Networks (COMSNETS)*, pages 1–8. IEEE, 2014.
- [33] Anton Barhan and Andrey Shakhomirov. Methods for sentiment analysis of twitter messages. In *12th Conference of FRUCT Association*, 2012.
- [34] Vasile-Daniel Păvăloaia, Elena-Mădălina Teodor, Doina Fotache, and Magdalena Danileț. Opinion mining on social media data: Sentiment analysis of user preferences. *Sustainability*, 11(16):4459, 2019.
- [35] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter sentiment analysis: The good the bad and the omg! In *Fifth International AAI conference on weblogs and social media*, 2011.
- [36] Hussam Hamdan, Frederic Béchet, and Patrice Bellot. Experiments with dbpedia, wordnet and sentiwordnet as resources for sentiment analysis in micro-blogging. In *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 455–459, 2017.