CENTRO UNIVERSITÁRIO CARIOCA

CURSO DE CIÊNCIA DA COMPUTAÇÃO

RÔMULO RODRIGUES COUTINHO

ANÁLISE DE SENTIMENTOS DO *TWITTER* EM RELAÇÃO A COVID-19 E A COMPARAÇÃO DE ALGORITMOS CLASSIFICADORES

RIO DE JANEIRO

2020

RÔMULO RODRIGUES COUTINHO

ANÁLISE DE SENTIMENTOS DO *TWITTER* EM RELAÇÃO A COVID-19 E A COMPARAÇÃO DE ALGORITMOS CLASSIFICADORES

Trabalho de conclusão de curso apresentado ao Centro Universitário Carioca, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Orientador(a): Prof.ª Daisy Cristine Albuquerque da Silva

RIO DE JANEIRO

2020

Coutinho, Rômulo Rodrigues.

Análise de sentimentos do twitter em relação a Covid-19 e a comparação de algoritmos classificadores. / Rômulo Rodrigues Coutinho. - Rio de Janeiro, 2020.

74f.

Orientadora: Dayse Cristine Albuquerque da Silva.

Trabalho de Conclusão de Curso (Graduação Superior em Ciência da Computação) - Centro Universitário Carioca, Rio de Janeiro, 2020.

1. Redes sociais. 2. Twitter. 3. Mineração de dados. I. Silva, Dayse Cristine Albuquerque da, prof. orient. II. Título.

CDD 005

RÔMULO RODRIGUES COUTINHO

ANÁLISE DE SENTIMENTOS DO *TWITTER* EM RELAÇÃO A COVID-19 E A COMPARAÇÃO DE ALGORITMOS CLASSIFICADORES

Trabalho de conclusão de curso apresentado ao Centro Universitário Carioca, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Rio de Janeiro, 02 de dezembro de 2020.

BANCA EXAMINADORA

_____

Professora Daisy Cristine Albuquerque da Silva. M. Sc. – Orientadora
Centro Universitário Carioca

_____

Professor Sérgio dos Santos Cardoso Silva. M. Sc. – Professor Convidado
Centro Universitário Carioca

_____

Professor André Luiz Avelino Sobral. M. Sc. – Coordenador de Curso
Centro Universitário Carioca

AGRADECIMENTOS

À Deus, pois sem ele nada é possível.

Aos meus pais, por todo o suporte e sacrifícios em prol de meu benefício.

À minha orientadora, por toda ajuda e paciência durante o desenvolvimento deste trabalho.

À Unicarioca, seu corpo docente e funcionários.

“Se a meta principal de um capitão fosse preservar seu barco, ele o conservaria no porto para sempre.”

São Tomás de Aquino

RESUMO

As redes sociais são uma ferramenta que proporciona a expansão das mídias, altera o comportamento e a realidade social da sociedade, permitindo uma grande ampliação de vozes, onde os usuários compartilham pensamentos, ideias e experiências. O compartilhamento de informações entre os usuários em um curto espaço de tempo, sobretudo relacionados a eventos que estão em alta no momento, geram um grande volume de dados, uma espécie de repositório dinâmico que podem ser úteis se forem tratados e analisados. Desta forma, em um momento em que o mundo vive uma pandemia de coronavírus, surge o estímulo de gerar conhecimento a partir destas informações, bem como, observar os métodos na computação que se mostram mais eficientes neste tipo de análise. O objetivo deste trabalho consiste em promover uma análise de sentimentos das mensagens da rede social *Twitter* relacionados a pandemia de covid-19 e, a partir disso, aplicar algoritmos classificadores e observar os respectivos valores de acurácia e eficiência dos mesmos. Os dados coletados do Twitter, *os tweets*, foram rotulados manualmente de acordo com o sentimento expresso entre “positivo”, “negativo” ou “neutro”, posteriormente passaram pelo processo de limpeza de dados, o pré-processamento, com o intuito de retirar *stopwords*, links e outros caracteres indesejáveis que não possuem valor semântico, bem como, é gerado uma nuvem de palavras mostrando os termos mais comuns entre os *tweets*. Após estes processos, com os dados devidamente “limpos” e rotulados é aplicado um total de seis algoritmos classificadores, são eles: Multinomial Naive Bayes, Bernoulli Naive Bayes, Complement Naive Bayes, Gaussian Naive Bayes, SVM - Support Vector Machines e Vizinhos Próximos (KNN). Em ordem de avaliar a performance dos respectivos algoritmos, é gerado seus valores de acurácia pelo método Predict com 70% da base de treinamento e 30% para teste, pelo método de validação cruzada com 10 páginas, bem como, é gerado a tabela de classificação contendo os resultados de precisão, revocação e *F-measure,* e por fim é gerado uma matriz de confusão contendo exatamente os números de classificações para cada classe.

Palavras chave: Redes Sociais, Twitter, Mineração de dados, Análise de Sentimentos, Classificação, Aprendizado de máquina.

ABSTRACT

Social networks are a tool that offers the expansion of the media, changes the human behavior and social reality of society, allowing a great expansion of voices, where users share thoughts, ideas and experiences. The sharing of information between users in a short period of time, especially related to events that are currently on the rise, generates a large volume of data, a kind of dynamic repository that can be useful if treated and compensated. Thus, at a time when the world is facing a coronavirus pandemic, there is a stimulus to generate knowledge from this information, as well as to observe the computational methods that are most efficient in this type of analysis. The objective of this research is to promote a sentiment analysis from Twitter messages related to the pandemic of covid-19 and, from that, apply classifying algorithms and observe their respective values of accuracy and efficiency. The data collected from Twitter, the tweets, were manually labeled according to the feeling expressed between “positive”, “negative” or “neutral”, later they went through the data cleaning process, the pre-processing, in order to remove stopwords, links and other undesirable characters that have no semantic value, as well, a word cloud is generated showing the most common terms among tweets. After these processes, with the data properly “cleaned” and labeled, a total of six classifying algorithms are applied, they are: Multinomial Naive Bayes, Bernoulli Naive Bayes, Complement Naive Bayes, Gaussian Naive Bayes, SVM - Support Vector Machines and Nearest Neighbors (KNN). In order of performance evaluation of the algorithms, their accuracy values are generated by the Predict method with 70% of the training base and 30% for testing, by the 10-page cross-validation method, and also a classification table is generated containing the results of precision, recall and F-measure , and finally a confusion matrix is generated with the exactly classification numbers for each class.

Keywords: Social networks, Twitter, Data mining, Sentiment Analysis, Classification, Machine Learning.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURA E SIGLAS

AI - Aprendizado indutivo

AM - Aprendizado de Máquina

APIs - Application Programming Interface

AS - Análise de Sentimentos

CSV - Comma-separated-values

DAU - Daily Active Users

SVM - Support Vector Machines

RNA - Ribonucleic acid

SMS - Short Message Service

MD - Mineração de Dados

OMS - Organização Mundial da Saúde

HTML - Hypertext Markup Language

IA - Inteligência Artificial

PLN - Processamento de Linguagem Natural

KNN - K Nearest Neighbors

SUMÁRIO

1. INTRODUÇÃO

Mediada pelos meios digitais, a sociedade vive uma ampliação de vozes, onde as pessoas estão aprendendo a compartilhar pensamentos, ideias e experiências através de novos modos de produzir e consumir conteúdo. A realidade se confunde com o virtual e vice-versa e o comportamento social se altera, perdendo suas amarras e abrindo espaço para a fluidez de informações, comportamento e relacionamentos (Rocha e Alves, 2010).

No cenário dos meios digitais, a ferramenta que proporciona a expansão das mídias e altera o comportamento e a realidade social da sociedade são as redes sociais, que podem ser definida por um conjunto de dois elementos: atores (pessoas, instituições ou grupos; os nós da rede) e suas conexões (interações ou laços sociais) (Wasserman e Faust, 1994; Degenne e Forse, 1999). Uma rede, assim, é uma metáfora para observar os padrões de conexão de um grupo social, a partir das conexões estabelecidas entre os diversos atores (Recuero, 2009).

Nos dias de hoje, as redes sociais são capazes de gerar quantidades massivas de dados, pessoas compartilham e expressam suas ideias sobre todos os assuntos e de uma maneira muito simples e instantânea (Gironés, 2020), tornando a web em um grande repositório dinâmico e nesse ambiente cotidiano de conectividade das redes é natural que em momentos de eleições, desastres e outros grandes acontecimentos, esses eventos sejam intensamente comentados e debatidos pelos usuários e um desses assuntos muito comentado é a COVID-19. (Filho e Coutinho, 2020)

O Coronavírus-2(SARS-CoV-2) é uma doença respiratória aguda grave, um vírus contagioso pertencente a uma família de vírus RNA positivo de cadeia simples conhecido como *coronavidae*, COVID-19 é uma sigla para “Doença Coronavírus 2019(Chamola; Hassija; Gupta; Guizani, 2020).

A pandemia do novo coronavírus é o maior desafio sanitário, econômico, social, humanitário e político do século 21, com mais de 37.423.660 casos confirmados em todo o mundo e mais de 1 milhão de mortes em 12 de outubro, no Brasil os números são: 5.082.637 casos confirmados e 150.198 mortes em 12 de outubro, segundo dados da OMS (World Health Organization – WHO).

Desse modo, a situação degradante provocada pela pandemia nas mais diversas esferas do convívio social e do comportamento humano, acarreta-se na expressão de múltiplos sentimentos e emoções humanas no ambiente de

conectividade das redes sociais, fomentando uma grande fonte de dados para estudos. Dentre as diversas plataformas de redes sociais, o Twitter está entre uma das mais populares com 186 milhões de usuários ativos *(DAU)* atualmente (Clement, 2020). O Twitter é uma rede social e um servidor para *microblogging* que possibilita aos seus usuários se expressarem em textos de até 240 caracteres *(tweets)* (Alencar; Rodrigues; Mendes; Peixoto, 2019), aumentando o limite para 280 caracteres em meados de 2018. Com o grande repositório de dados disponível após inúmeras postagens e interações dos usuários dentro da rede social, surge a necessidade de explora-los em busca de conhecimento, a plataforma Twitter disponibiliza algumas bibliotecas chamadas de *application programming interface (APIs)* para a extração dos dados da rede social, na qual será utilizada neste trabalho, especificamente a Twitter API Standard. Com isso, o objetivo é utilizar os dados capturados na rede social e implementar modelos de classificação de texto via aprendizado de máquina, com o intuito de avaliar e observar como se comportam diversos algoritmos classificadores na base de dados rotulada.

1.1 MOTIVAÇÃO E JUSTIFICATIVA

Em um momento único, onde o mundo sofre a maior crise sanitária do século, as redes socias se tornaram um campo fértil de dados, onde os usuários expressam seus sentimentos e como estão lidando com a pandemia e todo o caos que a mesma provoca, alia-se a isso, o fato da grande expansão do *Big data* e *Data Science,* onde o interesse pela informação e a necessidade de avaliação da mesma são cada vez mais demandada.

Sendo assim, a aplicação da análise de sentimentos é totalmente útil em várias áreas (Koblitz, 2020), devido as suas inúmeras aplicações para o ambiente corporativo, acadêmico e outros. Com isso, observar quais são os métodos na programação que se apresentam com maior eficácia em determinada análise se torna justificada.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo deste trabalho é observar como se comportam modelos classificadores a partir de sua aplicação em uma base rotulada. Os dados que compõe a base são provenientes dos usuários da rede social Twitter em relação a pandemia da COVID-19, verificando seus sentimentos expressos nos respectivos *tweets* em língua portuguesa. Logo, será feita a análise de sentimento e a utilização de técnicas de mineração de dados, como o pré-processamento de dados.

Os dados serão rotulados com seus determinados sentimentos - entre “positivo”, “negativo” ou “neutro”, serão aplicados e comparados diferentes métodos de *Machine Learning* e classificação de texto com os algoritmos Naive Bayes com quatro variações, SVM e KNN Vizinhos Próximos. É observado os respectivos valores de acurácia dos algoritmos, bem como é gerado as métricas de precisão, revocação, f-measure e a matriz de confusão, com o intuito de avaliar a performance dos mesmos.

Todos os procedimentos de extração, pré-processamento, e aplicação dos algoritmos classificadores foram feitos na linguagem *Python*, utilizando diversas bibliotecas disponíveis na mesma.

1.2.2 Objetivo Sumarizado

- Construir uma base de dados com os tweets capturados e fazer a rotulação manual do sentimento contido no mesmo – entre “positivo”, “negativo” ou “neutro”.
- Aplicar algoritmo classificadores Naive Bayes e suas variantes, SVM, e KNN nos dados da base construída.
- Observar a acurácia e outras métricas de avaliação do modelo gerado nos diferentes métodos aplicados.

1.3 ORGANIZAÇÃO DO TRABALHO

A partir de agora, este trabalho será organizado da seguinte maneira:

- **Capítulo 2 – Fundamentação Teórica**:

  Os conceitos da tecnologia, a revisão bibliográfica e toda a fundamentação teórica necessária para o perfeito entendimento do trabalho no decorrer do desenvolvimento e apresentação do documento, incluindo uma visão geral sobre Redes Sociais, Twitter, Mineração de Dados, Análise de Sentimentos, Aprendizado de Máquina, Classificação, bibliotecas e seus métodos e a avaliação do algoritmo.

- **Capítulo 3 – Desenvolvimento:**

  Neste capítulo é mostrado, de maneira detalhada, todos os processos feitos durante o desenvolvimento do trabalho, desde o processo de captura e extração dos tweets até a implementação dos algoritmos classificadores.

- **Capítulo 4 – Resultados:**

  Averiguação da implementação dos algoritmos de classificação, seus respectivos valores de acurácia demonstrada, bem como, são gerados uma tabela de classificação e a matriz de confusão.

- **Capítulo 5 – Considerações Finais:**

  Neste capítulo é feito um resumo do trabalho, seus objetivos e as principais afirmativas e deduções provenientes dos resultados do trabalho, bem como é apresentado sugestões de trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 REDES SOCIAIS

As redes sociais constituem uma das estratégias subjacentes utilizadas pela sociedade para o compartilhamento da informação e do conhecimento, mediante as relações entre atores que as integram (Tomaél; Alcará; Di Chiara, 2005)

A definição de Wasserman e Faust (1999) ajuda a compor uma definição de grafos da rede, onde os **atores**, termo vinculado a uma leitura mais sociológica, são os *Nós/Vértices*, que são representados pelas pessoas, grupos ou instituições e os *Laços* são as **relações** entre os atores e suas conexões e interações.

Segundo a definição de Duarte e Klaus (2008), uma rede social é uma estrutura social composta por pessoas ou organizações, conectadas por um ou vários tipos de relações, que partilham valores e objetivos comuns. Esse conceito de redes sociais começou a ser usado há cerca de um século. Após o surgimento da web e suas ferramentas de comunicação, esse termo passou a ser utilizado também para representar as ferramentas de comunicação e relacionamento que ampliaram as redes de contado dos usuários por meio da web (Araujo, 2014).

Tomaél, Alcará e Di Chiara (2005) detalham o comportamento sociológico do indivíduo nas redes, introduzindo a configuração organizacional gerada – Nas redes sociais, cada indivíduo tem sua função e identidade cultural. Sua relação com outros indivíduos vai formando um todo coeso que representa a rede. De acordo com a temática da organização da rede, é possível a formação de configuração diferenciadas e mutante.

Corroborando a ligação da instrumentação organizacional com o relacionamento humano introduzida pelas redes, Marteleto (2001) afirma que: “ ..[ ]..As redes sociais representam um conjunto de participantes autônomos, unindo ideias e recursos em torno de valores e interesse compartilhados, ressaltando que apenas nas últimas décadas o trabalho pessoal da rede de conexões passou a ser percebido como um instrumento organizacional, apesar de o envolvimento das pessoas em redes existir desde a história da humanidade.”

As redes sociais são ambientes facilitadores de criação e compartilhamento de conteúdo via web, que recentemente se tornaram parte do cotidiano. Alguns

ambientes virtuais populares nos dias de hoje são Twitter (twitter.com), Facebook (facebook.com), LinkedIn (linkedin.com), Instagram (instagram.com), WhatsApp (whatsapp.com). Estes ambientes possuem mecanismos que possibilitam aos usuários estabelecerem redes de relacionamento, sejam essas por afinidades profissionais, de amizade ou interesse em comum, além de possibilitar compartilhamento de conhecimento e opiniões sobre diversos temas. Dessa forma, o cenário proporcionou a disponibilização de uma enorme quantidade de dados e informações na web com opiniões e sentimentos dos usuários (Adaptado de - Araujo, 2014).

2.1.1 Números das Redes Sociais

Em 2020, estima-se que existam 3 bilhões de usuários ativos nas redes sociais, com expectativa de atingir 4.41 bilhões em 2025 (Clement, 2020).

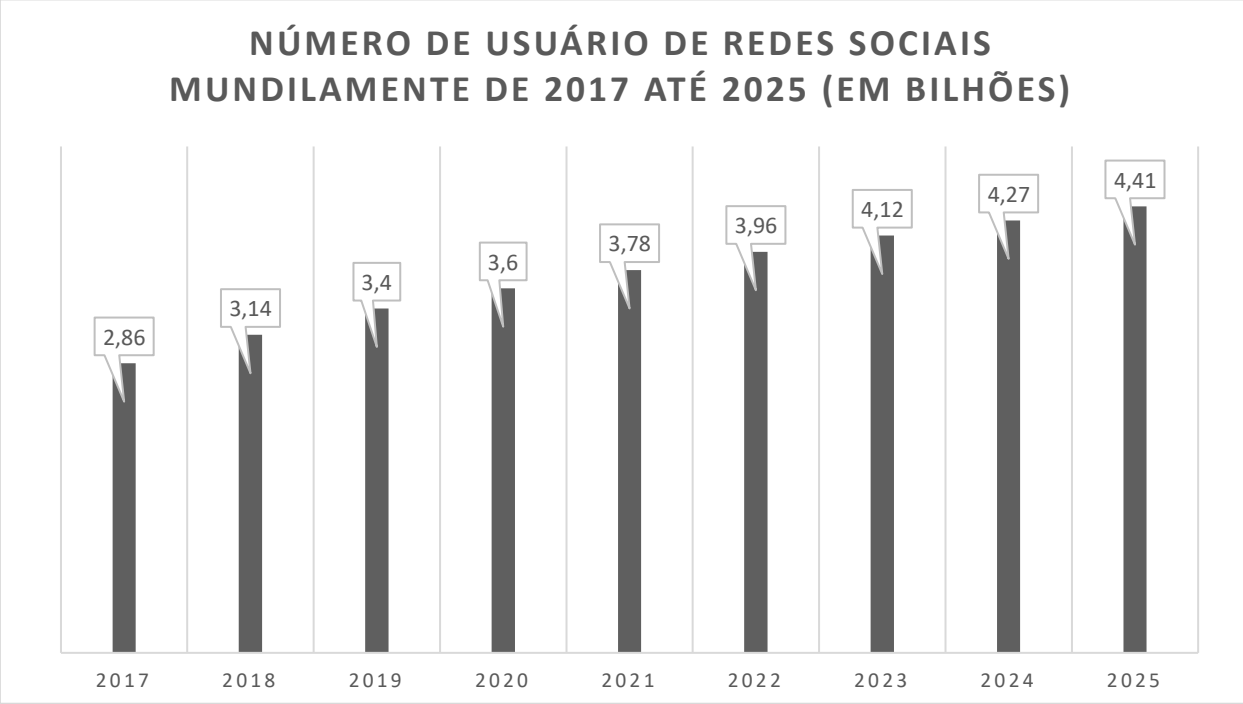

Figura 1. Número de usuário das redes sociais. Adaptado de Clement, 2020.

2.2 TWITTER

Segundo Lohse (2019), O Twitter é uma rede social que funciona como um servidor para microblogging, no qual os usuários enviam e recebem atualizações de outros contatos (os usuários que eles seguem), que devem se encaixar em textos de até 280 caracteres, denominados de *tweets*. Podem ser enviados por meio do *website* do Twitter, por SMS, por softwares de gerenciamento e pelo seu aplicativo. O Twitter foi criando em março de 2006 e lançado em 15 de julho de 2006, no seu lançamento a plataforma permitia apenas *tweets* com até 140 caracteres, modificando o limite para 280 caracteres em 2018, os proprietários do Twitter são Jack Dorsey, Evan Williams, Biz Stone e Noah Glass.

O Twitter permite ao usuário criar conteúdos diversos, pois uma de suas funcionalidades é a de realizar uma pergunta simples e direta na página inicial do sistema: “O que está acontecendo?”, com essa simples pergunta, o Twitter gera, diariamente, um grande volume de informações (Leite, 2015).

Segundo Castro (2011), O Twitter possui algumas caraterísticas importantes que o diferem de outras redes e criou tendências na maneira de engajamento e proliferação da informação, podemos citar as seguintes características:

- ***Tweets*** – Em português, to twit significa o ato das aves de gorjear, cantar notas rápidas, foi o que inspirou o nome da rede social: a emissão de sons curtos, os tuítes (*tweets*), como são chamados os textos publicados, tem um limite máximo de 280 caracteres.
- **Retweets** – Permite que o usuário replique um *tweet* existente para a sua lista de seguidores, podendo adicionar um comentário ou mídia “por cima” junto com a ação, sendo uma das principais ferramentas de difusão de conteúdo na rede social em questão.
- **Hashtags** – Hashtag é o símbolo “#” acompanhado de uma palavra, usado para repercutir um termo a um grupo de pessoas que o procura na busca (“Search”) do Twitter. A plataforma gera um link para cada hashtag, levando o usuário, caso seja clicado, a buscar os últimos resultados dos usuários que utilizaram a mesma hashtag.

- ***Followers e Following*** – A rede é composta pelos seus seguidores (Followers), que recebem automaticamente seus tuítes (*tweets*), e pelas pessoas que você segue (Following) para saber o que estão dizendo e compartilhando.

Segundo Araujo (2014), o Twitter oferece diversas bibliotecas chamadas de *application programing interface* (APIs), para tornar seu conteúdo amplamente disponível, em diferentes formatos e para outros sistemas, proporcionando as empresas e pesquisadores acadêmicos abertura para realizarem pesquisas e desenvolverem novas aplicações que envolvam essa rede social.

Este trabalho utiliza a Twitter API versão *Standard*, instanciada na linguagem *Python* para fazer requisições no servidor da aplicação com o objetivo de fazer a extração de *tweets* para montar uma base dados.

2.2.1 Números do Twitter

O Twitter é um fenômeno social relevante devido á sua grande quantidade de usuários e também de seu uso como dispositivo de marketing e camada de transporte para serviços de troca de mensagens de terceiros (Araujo, 2014).

Segundo o Washington Post (2020), o Twitter está vendo um número recorde de usuários migrar para o seu serviço em meio a pandemia do coronavírus.

Algumas estatísticas, segundo Lin (2020):

- Em média, um usuário fica logado na plataforma por 3:39 minutos por sessão.
- 500 milhões de *tweets* são enviados por dia.
- 75% de *B2B businesses* faz ações de marketing de seus produtos ou serviços na plataforma Twitter.

No gráfico a seguir, podemos observar o crescimento médio do número de usuários ativos mensais desde o primeiro trimestre do ano de 2015 até o primeiro trimestre do ano de 2019, onde atingiu a marca de 330 milhões (Clement, 2020).

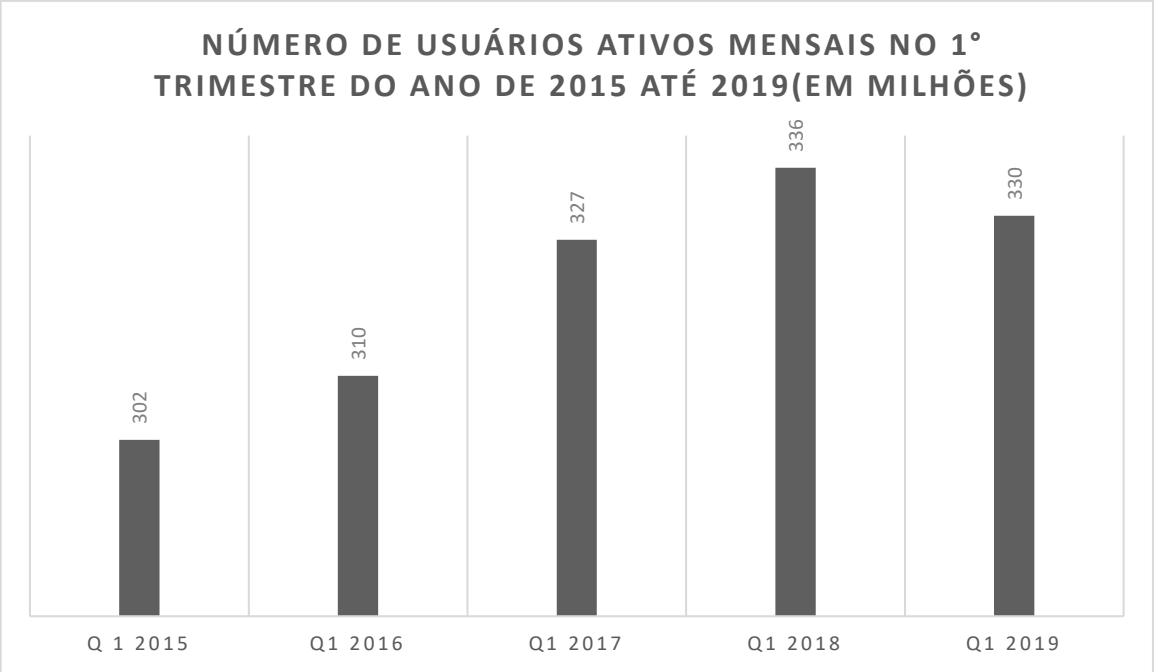

Figura 2. Número de usuários Twitter Q1 2015 - Q1 - 2019. Adaptado de Clement, 2020

No gráfico a seguir, podemos observar que os homens utilizam mais o Twitter em uma diferença de quase 2:1 em relação as mulheres (Lin, 2020).

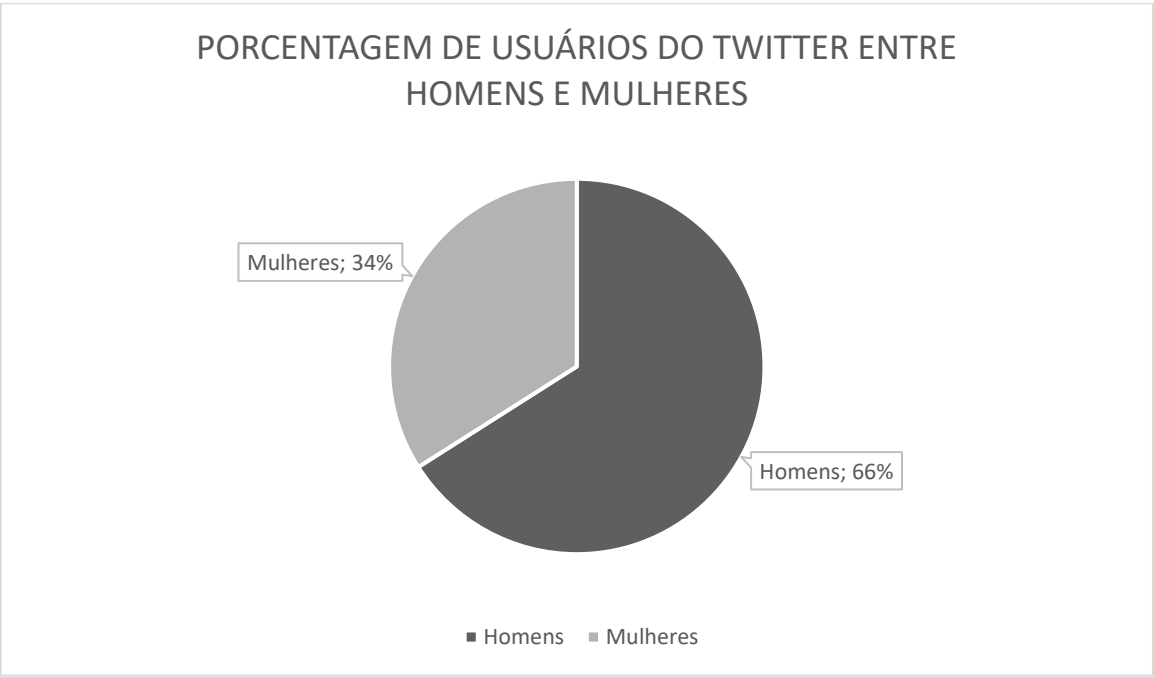

Figura 3. Divisão de Usuários Twitter entre Homens e Mulheres. Adaptado de Lin, 2020

2.3 MINERAÇÃO DE DADOS

Segundo Clement (2020), a população digital global de usuários ativos na internet atingiu 4.57 bilhões em julho, 2020 – número que, segundo Castro e Ferrari (2016), era de apenas 16 milhões em 1995. Nesse contexto que gerou uma superabundância de dados que surgiu a mineração de dados, como um processo sistemático, interativo e iterativo, de preparação e extração de conhecimentos a partir de grande base de dados.

Segundo Castro e Ferrari (2016 cap.1.2) Promovendo um comparativo com o processo clássico de mineração correspondente á extração de minerais valiosos, na mineração de dados (MD) se explora a base de dados (mina) usando algoritmos (ferramentas) adequadas para obter conhecimento (minerais preciosos). Os dados são símbolos ou signos não estruturados, sem significado, com valores em uma tabela, e a informação está contida nas descrições, agregando significado e utilidade aos dados, como o valor da temperatura do ar. Por fim, o conhecimento é algo que permite uma tomada de decisão para a agregação de valor.

Na análise de Amaral (2016), mineração de dados são processos para explorar e analisar grandes volumes de dados em busca de padrões, previsões, erros, associações entre outros. Normalmente a mineração de dados está associada ao aprendizado de máquina: Uma área da inteligência artificial que desenvolve algoritmos capazes de fazer com que o computador aprenda a partir do passado: usando dados de eventos que já ocorreram.

O aprendizado de máquina, continua Amaral (2016 pag.2), é capaz de identificar padrões que dificilmente seriam identificados a “olho nu” ou mesmo usando técnicas triviais de análise de dados, como filtros, pivôs ou agrupamentos. Na continuação deste trabalho, será aprofundado os conceitos de aprendizado de máquina, na qual, é parte central do objetivo da pesquisa sobre a eficácia de modelos classificadores em uma base rotulada.

Castro e Ferrari (2016) nos explica que os dados são símbolos não estruturados, logo, se faz necessário um processo para uma interpretação adequada, que assegura um conhecimento útil dos dados trabalhados. Fonseca e Araújo Jr (2018), detalham esse processo.

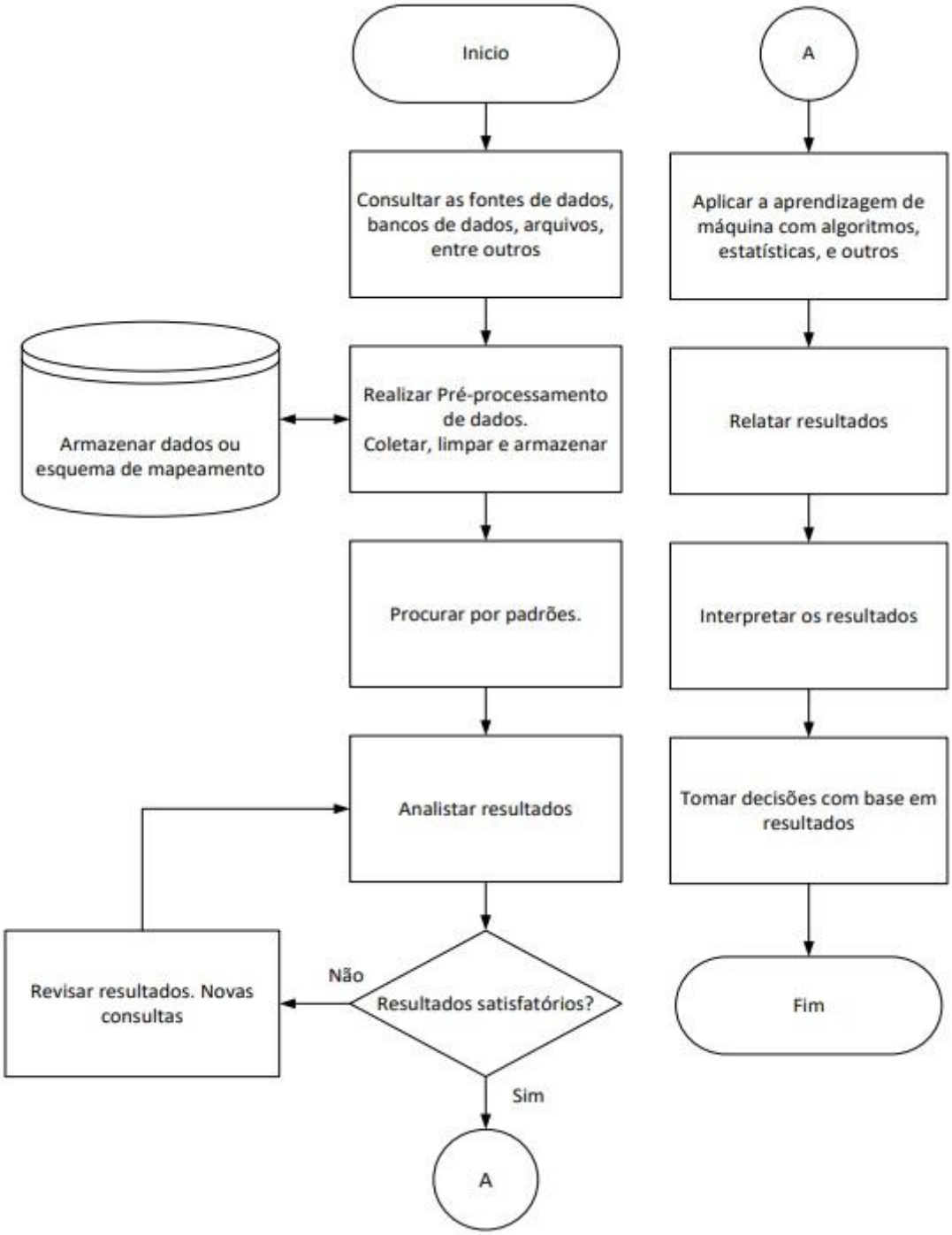

Figura 4. Fluxograma do processo de mineração de dados. Fonte: Fonseca e Araújo JR (2018)

2.4 MINERAÇÃO NA WEB

Segundo Marinho e Gerardi (2003), a Web é uma vasta coleção de documentos heterogêneos. Possui natureza dinâmica e milhões de páginas surgem e desaparecem todos os dias. Por isso sente-se um anseio para que a Web realmente alcance todo o seu potencial e se torne uma ferramenta mais utilizável, eficaz e compreensível. Nesse contexto a mineração de dados parece como uma possibilidade óbvia a ser explorada. Em parte pelo seu grande sucesso quando aplicada a banco de dados tradicionais, e em parte porque a Web parece ser uma área fértil em potencial para a aplicação de suas técnicas.

A necessidade de absorver essas informações é alta e sabendo-se o quão custoso é realizar tal tarefa através de meios não computacionais, pesquisadores desenvolveram o conceito de *web mining* / mineração web, derivado da expressão *data mining* / mineração de dados (Santos, L 2010).

Entretanto, Marinho e Girardi (2003), destacam uma grande diferença entre mineração de dados e mineração na web, sobretudo quanto a natureza disponibilizada dos dados – Utilizar e compreender os dados disponíveis na web não é uma tarefa simples, são muito mais sofisticados e dinâmicos do que os sistemas de armazenamento tradicionais. Enquanto esses últimos utilizas estruturas de armazenamento bem definidas e estruturadas, a web não possui qualquer controle sobre a estrutura ou o tipo dos documentos que virtualmente armazena. Corroborando com a explanação acima, Santos, R (2009) expõe que os armazenamentos tradicionais da mineração de dados se dão através de tabelas organizadas em linhas (dados) e colunas (atributos). Dados disponíveis na Web raramente seguem este padrão e diferentes categorias de dados tem diferentes padrões, e devem ser processados para uso com os algoritmos tradicionais de mineração de dados.

Na explanação de Cunico e Foppa (2016), as tarefas principais da mineração na web, que são subtarefas da mineração de dados, são as seguintes:

- Busca de documentos: Consiste em encontrar sites na Web contendo documentos específicos por palavra-chave. É o processo de extrair dados a partir de fontes de textos disponíveis na internet como conteúdo HTML.

- Seleção e pré-processamento da informação: Consiste em selecionar automaticamente informações obtidas na internet.
- Generalização: Consiste em descobrir padrões gerais em sites Web ou vários sites. Esta técnica envolve técnicas de Inteligência Artificial e Mineração de Dados.
- Análise: Validação e interpretação dos padrões minerados.

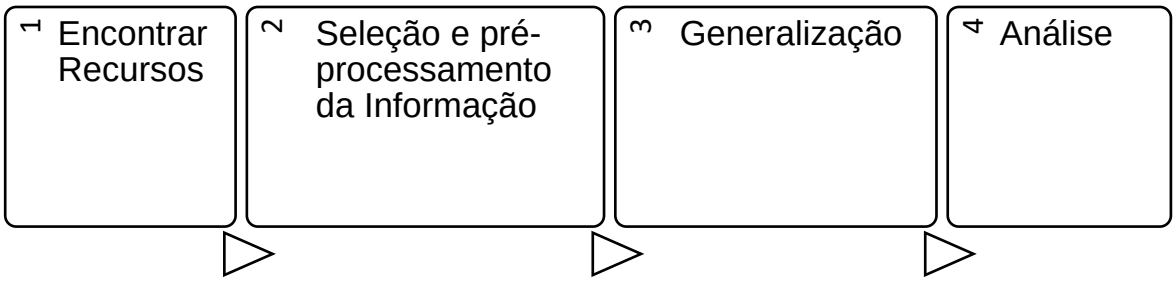

Figura 5. Sequência de subtarefas da Mineração de Dados. Adaptado de Guimarães, 2018.

A mineração da Web agrupa em três diferentes abordagens um conjunto de ferramentas importantes que além de descobrir as fontes de informações relevantes, pretende mapear e analisar o padrão de acesso e armazenamento de informações na web (Junior, 2007)

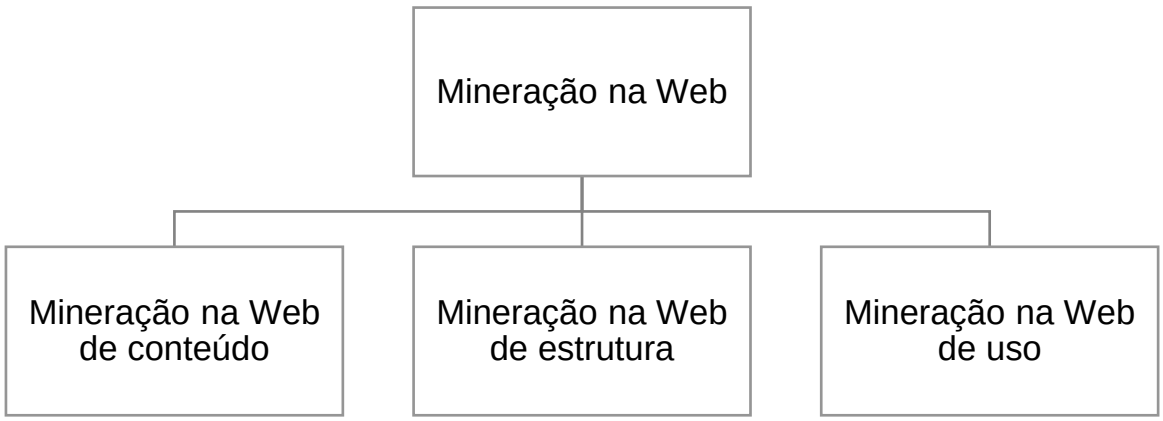

Figura 6. Categorias da Mineração da Web. Adaptado de Guimarães, 2018.

- **Mineração na Web de Conteúdo:** Consiste em analisar textos, imagens e outros componentes presentes nos documentos HTML. Esta técnica é essencialmente utilizada como forma de facilitar o acesso ao conteúdo predominante desestruturado encontrado nestes tipos de documento.
- **Mineração na Web de estrutura**: Estuda o relacionamento entre as páginas da web através de seus hyperlinks, com o objetivo de identificar páginas pertinentes a uma determinada área de conhecimento.
- **Mineração na Web de uso**: É a descoberta de conhecimento através do registro de visitação e de busca de usuários entre os diferentes sites na Internet. Desta forma, é possível identificar padrões de acesso, requisito essencial para, por exemplo, implementar o processo de personalização de uso que permite a utilização de um contexto próprio na busca de documentos na Internet, gerando resultados também personalizados.

Este trabalho está posicionado na categoria de Mineração na Web de Conteúdo, ou *web contente mining*, dado que, será analisado conteúdo de texto da Web – especificamente os *tweets* dos usuários da rede social Twitter em relação a pandemia da COVID-19.

2.5 MINERAÇÃO DE TEXTO

Mineração de textos, também chamado de mineração de dados textuais ou descoberta de conhecimento de base de dados textuais é um campo novo e multidisciplinar que inclui conhecimentos de áreas como Informática, Estatística, Linguística e Ciência cognitiva. Mineração de textos consiste em extrair regularidades, padrões ou tendências de grandes volumes de textos em linguagem natural, normalmente, para objetivos específicos (Aranha e Passos, 2006).

Segundo Morais e Ambrósio (2007), mineração de textos (*Text Mining*) é um Processo de Descoberta de Conhecimento, que utiliza técnicas de análise e extração de dados a partir de textos, frases ou apenas palavras. Envolve a aplicação de algoritmos computacionais que processam textos e identificam informações úteis e implícitas, que normalmente não poderiam ser recuperadas utilizando métodos

tradicionais de consulta, pois a informação contida nesses textos não pode ser obtida de forma direta, uma vez que, em geral, estão armazenados em formato não estruturado.

Ainda segundo Aranha e Passos (2006), o crescimento do armazenamento de dados não estruturados, devido ao avança da mídia digital, proporcionou o desenvolvimento das técnicas de mineração de textos. Normalmente, os documentos onde são aplicadas as técnicas de mineração de textos incluem: e-mails, textos livres obtidos por resultados de pesquisas, arquivos, gerados por editores de textos, páginas da Web, campos textuais em banco de dados, documentos eletrônicos, digitalizados a partir de papéis.

Figueiredo, Catini e Mendes (2018), apontam que técnicas de mineração de texto, quando aplicadas ao conteúdo publicado pelos usuários da Web, oferecem processos para obtenção de informações importantes de textos. Porém, como não existe padronização na forma como as pessoas se expressam na Internet, existem inúmeros desafios na aplicação das técnicas de mineração.

Segundo Cheng (2016), a maioria dos problemas na mineração de texto se concentra em duas partes:

i. Extração e seleção de recurso;
ii. Métodos de aprendizagem de máquina para classificação.

Aranha e Passos (2006) detalham quais são os processos de mineração textos na figura a seguir:

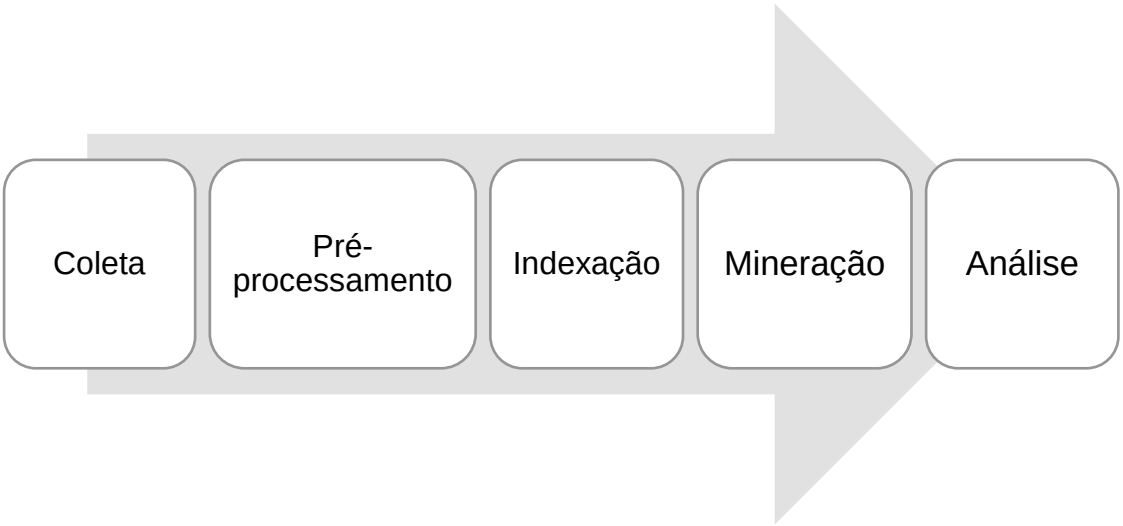

Figura 7. Processo de Mineração de Textos - Adaptado de Aranha e Passos, 2006.

A primeira etapa a ser realizada é a de **Coleta**, cujo objetivo é a formatação da coleção de documentos, elemento básico de qualquer processo de Mineração de Textos.

Em seguida, inicia-se a etapa de **Pré-processamento**. É neste momento que os documentos, obtidos na fase anterior, são submetidos a inúmeras operações capazes de obter uma forma de representa-los estruturadamente.

Após o Pré-Processamento, inicia-se a fase de **Indexação**. Indexação é o processo responsável pela criação de estruturas auxiliares denominadas índices e que garantem rapidez e agilidade na recuperação dos documentos e seus termos.

Uma vez indexados, os documentos são submetidos a algoritmos de Aprendizado de Máquina e de Estatística para que seja realizada a extração de conhecimento dos mesmos. A extração de conhecimento tem a finalidade de descobrir padrões úteis e desconhecidos presentes nos documentos.

E finalizando um processo de Mineração de Textos, temos a etapa de **Análise**. Na etapa de análise é realizada a avaliação e intepretação de todo o conhecimento obtido pelo processo.

Segundo Gonçalves (2012), uma enorme quantidade de textos encontrados na Internet – blogs, redes sociais, etc. – reflete a opinião de pessoas a respeito de algum produto, serviço, programa de TV, jogo de futebol, filme, livro, discurso político, etc. A análise de sentimentos é uma nova tarefa de mineração de textos que tem por objetivo identificar a opinião, emoção e sentimento das pessoas sobre um determinado tema, a partir da análise de textos. Por esta razão, a tarefa também é comumente referenciada como *opnion mining*. No decorrer deste trabalho será aprofundado sobre a análise de sentimentos, que é uma das tarefas de mineração de textos, como definido antes no objetivo do trabalho, será feita uma análise de sentimentos dos textos da rede social Twitter.

2.6 ANÁLISE DE SENTIMENTOS

A análise de sentimentos, também chamada de análise ou mineração de opinião, ou de computação afetiva, tenta classificar textos atribuindo a ele uma orientação, a qual pode ser positiva, negativa ou neutra (Koblitz, 2010). Segundo Figueiredo, Catini e Mendes (2018), a análise de sentimentos é uma área da mineração de dados que se utiliza de técnicas de processamento de linguagem natural (PLN/NLP – *Natural Language Processing*) e mineração de textos para classificar a polaridade da opinião. As técnicas de processamento de língua natural (PLN) analisa o “corpus” do texto reconhecendo o contexto da informação, ou seja, analisando por meio de viés, na análise de sentimento a PLN atua em conjunto com a mineração de texto.

Segundo Koblitz (2010), a palavra sentimento define o que uma pessoa sente a respeito de algo, pode ser também uma atitude mental de aprovação ou não a respeito de um determinado assunto, ou mesmo pode ser uma opinião ou uma reflexão.

Liu (2010) define AS como um campo de estudo onde analisa opiniões, sentimentos, avaliações, atitudes e emoções para entidades, por exemplo, produtos, serviços, organizações, indivíduos, eventos, tópicos e seus atributos.

Koblitz (2010) afirma que o objetivo da análise de sentimentos é entender como o leitor pode interpretar uma emoção em um texto e com isto desenvolver programas que executem esta tarefa.

Certos tipos de emoções quando expressas através de ironias ou metáforas estão fora do escopo desta análise por serem muito complexas.

Ainda segundo Koblitz (2010), a análise de sentimentos envolve a identificação de:

a) Expressões ou palavras que expressam sentimentos;
b) A polaridade (positivo/negativo/neutro) e intensidade das expressões;
c) O relacionamento destas com o assunto examinado.

2.6.1 Etapas da AS

Matioli (2010) detalha as fases envolvidos no processo de Análise de Sentimentos, estas etapas são: Coleta de dados, Processamento, Analise e Apresentação de Resultado.

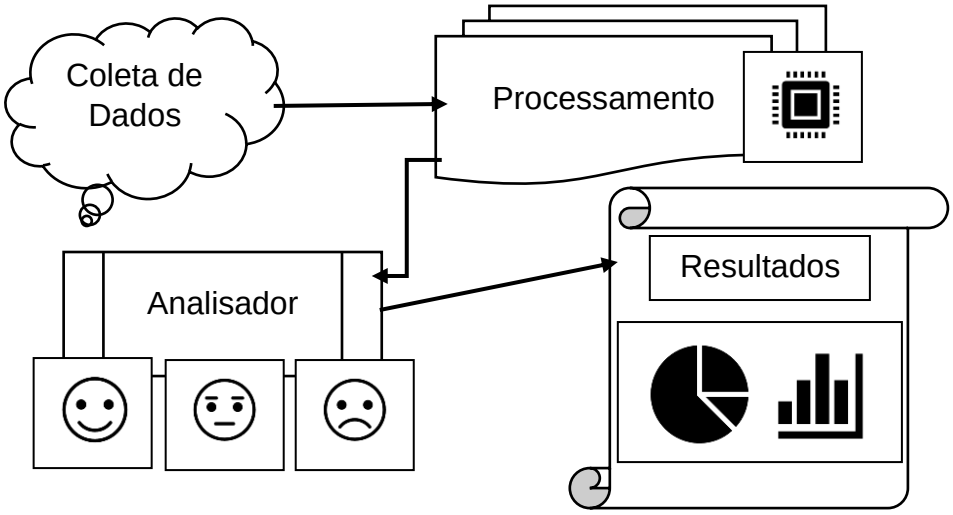

Figura 8. Fases do Processo de AS. Adaptado de Matioli, 2010

- **Coleta de dados:** Define-se qual a fonte de dados será utilizada, a qual pode ser textos das redes sociais, web, documentos e etc.
- **Preparação dos dados:** Conhecida também como pré-processamento, é a etapa onde os dados passarão por um tratamento, para se corrigir possíveis problemas com erros ortográficos, abreviaturas, gírias, ditos populares, além da tradução de comentários.
- **Classificação dos sentimentos:** É a etapa principal de um sistema de AS, pois é nessa fase que são aplicadas técnicas de análise e os textos são classificados como positivos, negativos ou neutros.
- **Sumarização de resultados**: Nessa etapa os resultados tem que ser exibidos de forma clara para que o usuário possa entender. Geralmente os resultados são exibidos em formas de gráfico ou textual, porém apresentá-los em forma textual pode deixar o usuário confuso devido a grande quantidade de textos,

por isso é indicado utilizar gráficos, pois são de caráter estatístico, oferecendo uma melhor compreensão.

2.6.2 Desafios da AS

Rodrigues, Vieira, Malagoli, Timmermann, apresentam mais detalhadamente os desafios da análise de sentimentos:

- Textos com erros e sentenças sintaticamente mal formadas (o que é bastante comum nos blogs e redes sociais) dificultam a busca e classificação dos mesmos;
- Não é trivial distinguir se um texto é opinião ou fato, e principalmente em um fato se existem opiniões embutidas;
- Textos podem conter sarcasmos e ironias, que são difíceis de serem identificados e podem impactar os resultados;
- Um texto pode referenciar mais de um item de interesse com opiniões diferentes sobre os itens, o que pode confundir a classificação;
- Uso de pronomes para referenciar itens pode dificultar a identificação de sentenças que mencionam o item de interesse;
- Uso de termos informais e gírias da internet devem ser considerados no vocabulário;
- Propaganda disfarçada, em que blogueiros recebem dinheiro para falar bem de alguma empresa ou produto pode impactar os resultados.

2.6.3 Desafios da AS no Twitter

Neste trabalho, a análise de sentimentos foi realizada através de mensagens capturadas na rede social Twitter, e a mesma compartilha diverso dos desafios sumarizados acima, pois, segundo Corrêa (2017), é uma rede que encoraja a publicação de textos curtos e, por isso, dificuldades podem surgir durante o processamento dos *tweets,* algumas relevantes neste ambiente são: Ambiguidades, Sarcasmos, Variações da ortografia e repetição de letras para enfatizar sentimento.

2.7 APRENDIZADO DE MÁQUINA

Aprendizado de Máquina é a ciência (e a arte) da programação de computadores para que eles possam aprender com os dados (Géron, 2019).

Segundo Monard e Baranauskas (2003), Aprendizado de Máquina é uma área de IA cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado bem como a construção de sistemas capazes de adquirir conhecimento de forma automática. Um sistema de aprendizado é um programa de computador que toma decisões baseado em experiências acumuladas através da solução bem sucedida de problemas anteriores. Os diversos sistemas de aprendizado de máquina possuem características particulares e comuns que possibilitam sua classificação quanto á linguagem de descrição, modo, paradigma e forma de aprendizado utilizado.

Diz-se que um programa de computador aprende pela experiência E em relação a algum tipo de tarefa T e alguma medida de desempenho P se o seu desempenho em T, conforme medido por P, melhora com a experiência E. (Mitchell, 1997 – citado por Géron, 2019).

Brunialti, Freire, Peres e Lima (2015), explica que a teoria de AM é baseada nos princípios do aprendizado indutivo (AI), ou seja, modelos são determinados a partir de um conjunto de dados ou representações de experiências. Normalmente, o aprendizado indutivo é implementado por algoritmos que processam um conjunto de dados e extraem um modelo capaz de explicar ou representar os dados sob algum aspecto. Esse modelo pode ser usado para explicar ou representar um novo dado (do mesmo domínio do conjunto inicial).

Sobre o AI, Monard e Baranauskas (2003), afirma que a indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos. Ela é caracterizada como raciocínio que se origina em um conceito específico e o generaliza, ou seja, da parte para o todo. Na indução, um conceito é aprendido efetuando-se inferência indutiva sobre os exemplos apresentados. Portanto, as hipóteses geradas através da inferência indutiva podem ou não preservar a verdade. Mesmo assim, a inferência indutiva é um dos métodos utilizados para derivar conhecimento novo e predizer eventos futuros.

O AI pode ser de três modalidades: supervisionado, não-supervisionado e semi-supervisionado, na tabela a seguir (*Tabela 1*), é feito a descrição sumarizada das respectivas modalidades de AI.

Modalidade	Descrição
Supervisionado	Os algoritmos ajustam parâmetros de um modelo a partir do erro medido entre respostas obtidas e esperadas.
Não supervisionado	Os parâmetros de um modelo são ajustados com base na maximização de medidas de qualidade das respostas obtidas.
Semi supervisionado	É caracterizado pelo uso de algoritmos híbridos, que fazem uso dos recursos de correção de erro e de maximização de medidas de qualidade, conforme necessário.

Tabela 1. Modalidades de Aprendizado Indutivo - Adaptado de Brunialti, Freire, Peres e Lima (2015)

Aplicações

O Aprendizado de Máquina possui diversas aplicações no estudo de dados, relacionadas ao setor corporativo, segundo Géron (2019), podemos citar os seguintes exemplos práticos:

- Segmentar clientes e encontrar a melhor estratégia de marketing para cada grupo;
- Recomendar produtos para cada cliente com base no que clientes similares compraram;
- Detectar quais transações são susceptíveis de serem fraudulentas;
- Prever a receita do próximo ano.

Neste trabalho, apresentamos os conceitos da aprendizagem de máquina, bem como suas modalidades de AI, e através do método de classificação (que entraremos em detalhes a seguir), via método supervisionado, a aplicabilidade em questão se focará na construção de modelos de aprendizagem de máquina para que seja

averiguada os seus respectivos valores de acurácia na base rotulada. O aprendizado de máquina, é implementado através de algoritmos classificadores.

2.8 CLASSIFICAÇÃO

Os algoritmos de Classificação realizam a predição de categorias. Os mais conhecidos são árvores de decisão, redes bayesianas e os vizinhos mais próximos (Leite,2015). A técnica de classificação tenta prever a classe do objeto representado por uma instância baseada nos valores de seus atributos. Segundo vários autores, este processo de classificação é uma das técnicas possíveis de aprendizado de máquina (Goldschimdt e Passos, 2005)

Nesse sentido, os algoritmos de classificação são usados para delegar uma tarefa de atribuição de objetos a uma das categorias pré-definidas, eles definem a classificação como uma técnica de mineração de dados que está na categoria de aprendizagem supervisionada.

Segundo Santos (2013), a classificação consiste em atribuir uma classe a um documento a partir dos seus dados de entrada. Essa classificação é feita através de algoritmos de aprendizagem. Esses algoritmos conseguem “criar conhecimento” sobre o domínio tratado através de dados de entrada. Quando esses dados de entrada já possuem suas classes atribuídas dizemos que essa é uma aprendizagem supervisionada. Dessa forma, o algoritmo consegue criar uma relação entre as características de cada entrada com a sua classe final, gerando um “conhecimento” para fazer previsões sobre entradas que ainda não possuem uma classe definida. Esses dados de entrada já classificados são chamados de base de treino enquanto que os documentos que serão classificados posteriormente ao treino são denominados como a base de teste.

Para se executar a tarefa de classificação, são usados dados que consistem em um conjunto de atributos denominados previsores, e um atributo denominado preditor (classe), que são definidos na tabela a seguir (*Tabela 2*).

Atributos	Descrição
Previsores	São utilizados para definir uma classificação efetiva dos registros pertencentes á base de dados em estudo.
Preditor (classe)	É utilizado como uma hipótese de classificação que será válida ou não pela análise resultante da classificação através dos atributos previsores.

Tabela 2. Atributos Previsores e Preditor, Adaptado de Carvalho, 2001.

Incialmente, o conjunto de treinamento é percorrido, analisando as relações existentes entre os atributos previsores e o atributo preditor. Estas relações são então usadas para prever a classe dos registros presentes no conjunto de teste, que será a próxima ação do classificador (Mitchell, 1997). Segundo Monard e Baranauskas (2001), em um segundo momento quando o algoritmo analisará o conjunto de teste, o atributo preditor não é considerado. Após a previsão das classes dos registros do conjunto de teste, essas classes são comparadas com as classes da hipótese definida pelo atributo preditor. Com isso pode-se comparar o número de previsões corretas e incorretas.

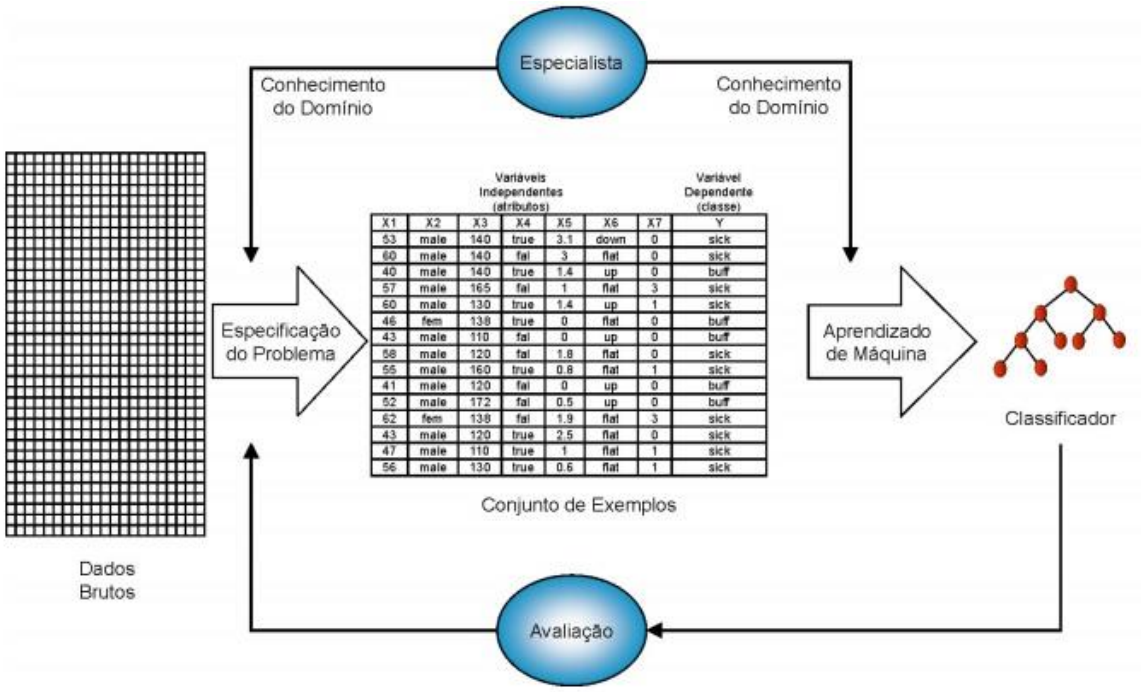

Figura 9. Processo de Classificação, Fonte: Monard e Baranauskas, 2003.

De maneira geral, o processo de classificação pode ser ilustrado pela Figura 9 na página anterior.

O conhecimento sobre o domínio pode ser usado para escolher os dados ou para fornecer alguma informação previamente e o processo de classificação pode ser repetido, se necessário, por exemplo, adicionando outros atributos, exemplos ou mesmo ajustando alguns parâmetros no processo de IA (Monard e Baranauskas, 2003).

2.8.1 Naive Bayes

Os algoritmos classificadores de documentos utilizam processos indutivos. Nesta linha, como explicado na seção 2.6 (Classificação), um classificador para uma categoria c_i é construído observando as características de um conjunto de documentos, previamente rotulados sob c_i por um especialista no domínio. Esta é uma abordagem de aprendizado supervisionado, onde um novo documento é classificado de acordo com as características aprendidas por um classificador construído e treinado a partir de dados rotulados (Martins, 2003).

Neste trabalho, após a coleta dos dados, rotulação e pré-processamento do mesmo, foi aplicado de diferentes formas e com quatro variações do algoritmo Naive Bayes pela biblioteca *Scikit-Learn* da linguagem *Python.*

Naive Bayes é um algoritmo probabilístico de classificação relativamente simples. O algoritmo se baseia no teorema de Bayes e recebe o nome Naive (ingênuo) por assumir algumas fortes hipóteses de independência condicional entre as variáveis (Santos, 2013).

Segundo Zhang (2004), o teorema de Bayes afirma a seguinte relação, dada a variável de classe e veto de características dependentes através de: $yx_1 x_n$

Mostrado na figura a seguir, Figura 10.

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Figura 10 - Teorema de Bayes

Em ordem de diminuir a quantidade de parâmetros calculados, assumindo algumas hipóteses, que apesar de incorretas, simplificam o modelo e ainda assim nos geram resultados satisfatórios (Santos, 2013).

1. Assume-se que a posição em que as palavras aparecem no texto não importa;
2. Assume-se que as probabilidades de cada atributo são independentes da condicional dada.

Os diferentes classificadores Naive Bayes diferem principalmente pelas suposições que fazem em relação à distribuição de:

$$P(x_i \mid y)$$

Apesar de suas suposições aparentemente super simplificadas, os classificadores Naive Bayes têm funcionado muito bem em muitas situações do mundo real, podemos citar a famosa classificação de documentos e filtragem de *spam*. Eles requerem uma pequena quantidade de dados de treinamento para estimar os parâmetros necessários.

Os classificadores Naive Bayes podem ser extremamente rápidos em comparação com métodos mais sofisticados. A dissociação das distribuições de características condicionais da classe significa que cada distribuição pode ser estimada independentemente como uma distribuição unidimensional. Isso, por sua vez, ajuda a aliviar os problemas decorrentes da maldição da dimensionalidade. Neste trabalho, foi aplicado quatro métodos do Naive Bayes pela biblioteca *Scikit-Learn* para averiguar os resultados de avaliação dos mesmos.

2.8.2 SVM – Support Vector Machine

O SVM, ou Support Vector Machine, é um algoritmo de aprendizado de máquina supervisionado, usado principalmente em problemas de classificação binária, podendo ser utilizado em classificação com múltiplas classes ou regressão. O classificador SVM é amplamente utilizado em tarefas de classificação de texto devido ao seu alto desempenho obtido em comparação com demais métodos e robustez para lidar com *outliers (*leia-se*,* dados que se diferenciam drasticamente de todos os outros, são pontos fora da curva normal) e alta dimensionalidade (Dorneles, 2019).

O objetivo de um SVM é encontrar o hiperplano de separação ideal, o qual maximiza a margem da base de treinamento (Câmara, 2015). Na Figura 11, é representado esse processo do SVM.

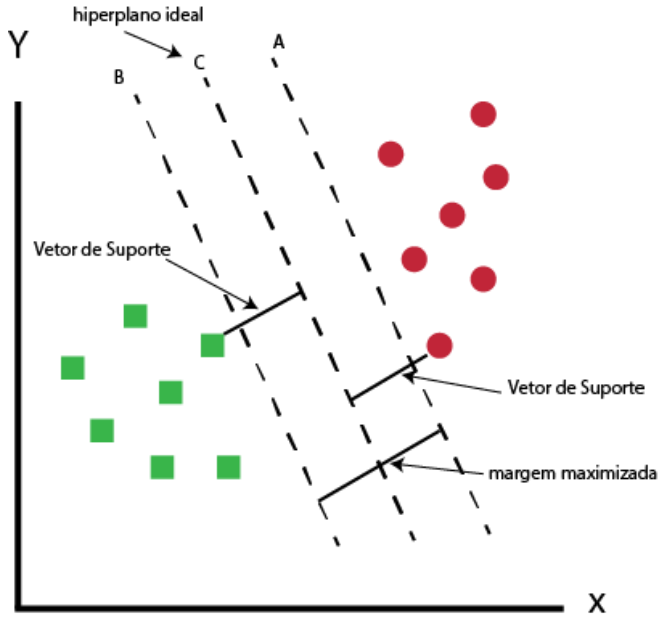

Figura 11 SVM - Hiperplano ideal

No exemplo da Figura 11, dado as classes representadas em verde e outras em vermelho, o algoritmo SVM propõe um hiperplano ideal entre as mesmas, com o objetivo de maximizar a margem de base de treinamento.

2.8.3 Vizinhos Próximos (*KNN*)

A ideia principal do classificador KNN é determinar o rótulo de classificação de uma amostra baseado nas amostras vizinhas de um conjunto de treinamento. O funcionamento do algoritmo ocorre da seguinte forma: Dado um problema de classificação com dois rótulos de classe e com K = x (é definido um valor de distância dos “vizinhos”), A variável K representa a quantidade de vizinhos mais próximos que serão utilizados para averiguar de qual classe a nova amostra pertence (*Pacheco, 2017).*

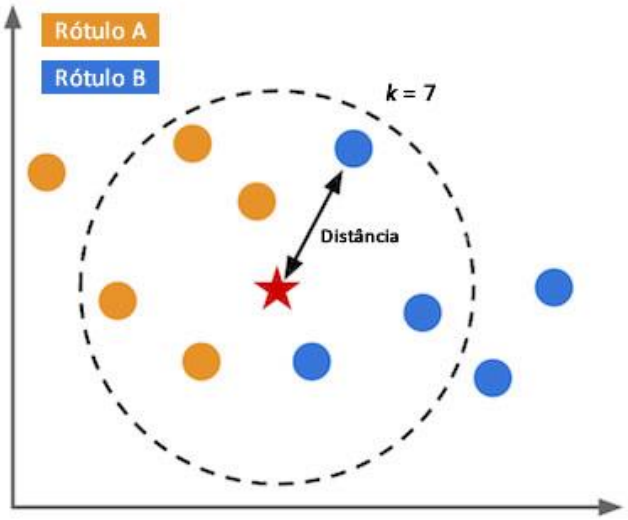

Figura 12. Exemplo de classificação KNN – com duas classes e k=7, Fonte: Pacheco, 2017

No exemplo da Figura 12, dado K = 7, nota-se que as 7 amostras mais próximas da nova amostra, 4 são do rótulo A e 3 do rótulo B. Portanto, como existem mais vizinhos do rótulo A, a nova amostra receberá o mesmo rótulo deles, ou seja, A (*Pacheco, 2017).*

Na aplicação do algoritmo KNN, é necessário determinar a métrica de distância e o valor do K. Neste trabalho foi definido K = 10, ou seja, ele analisará as 10 amostras mais próximas das novas amostras para definir a nova rotulação.

2.8.4 Biblioteca Scikit-Learn e os métodos classificadores

Em programação, bibliotecas são um conjunto de funções pré-escritas por desenvolvedores, assim permitindo que outros programadores possam utilizá-las, proporcionando um desenvolvimento mais rápido e fácil (Zanette, 2017).

No desenvolvimento deste trabalho foi utilizado diversas bibliotecas da linguagem *Python*, desde a extração dos tweets até a implementação dos algoritmos classificadores. A biblioteca *Scikit-Learn,* que é uma biblioteca de código aberto e licença livre, foi utilizada para a implementação destes algoritmos classificadores Naive Bayes, SVM e Vizinhos Próximos, através dos seguintes métodos:

1. MultinomialNB()
2. BernoulliNB()
3. ComplementNB()
4. GaussianNB()
5. Svm.SVC()
6. KneighborsClassifier()

Multinomial Naive Bayes

Segundo Singh, Kumar, Gaur e Tyagi (2019), o classificador Multinomial Naive Bayes trabalha com o conceito de frequência de termo, que significa quantas vezes a palavra ocorre em um documento. Este modelo informa dois fatos, independentemente de a palavra ocorrer em um documento ou não, bem como sua frequência nesse documento.

Considerando o fato que um termo específico pode ser primordial na decisão do sentimento de um documento, essa propriedade desse modelo o torna uma decente opção para classificação de documentos. Também, a frequência de um termo em um documento pode ajudar a determinar se um termo específico é importante em uma análise ou não (Singh, Kumar, Gaur e Tyagi, 2019)

Segundo a biblioteca Scikit-Learn, o método MultinomialNB implementa o algoritmo Naive Bayes para dados distribuídos de maneira multinominal, e é uma das variantes clássicas do Naive Bayes utilizado na classificação de texto.

Bernoulli Naive Bayes

O classificador Bernoulli Naive Bayes trabalha com o conceito binário de que se o termo ocorre em um documento ou não, ao contrário do Multinomial Naive Bayes, ele não utiliza a contagem de frequência do termo. Ao definir um modelo Bernoulli com os dados dos tweets e seus sentimentos rotulados, ele vai multiplicar todas as ocorrências de todas as palavras contidas nos tweets e também as probabilidades de não ocorrência de palavras que não ocorrem nos tweets (Singh, Kumar, Gaur e Tyagi, 2019).

Na definição da biblioteca Scikit-Learn, o método BernoulliNB implementa os algoritmos Naive Bayes de treinamento e classificação para dados distribuídos de acordo com as distribuições multivariadas de Bernoulli, ou seja, pode haver múltiplas características, mas cada uma delas é considerada uma variável de valor binário (Bernoulli = booleano).

Portanto o classificador Multinomial Naive Bayes é sobre a frequência que os termos se repetem no documento, enquanto o classificador Bernoulli Naive Bayes é sobre se o termo está presente ou não no documento.

Complement Naive Bayes

O classificador Complement Naive Bayes é particularmente adequado para trabalhar com conjunto de dados desequilibrados. O Complement NB, em vez de calcular a probabilidade de um item pertencente a uma determinada classe, se calcula a probabilidade do item pertencente a todas as classes. (Geeksforgeeks, 2016)

O classificador Complement NB é indicado para conjunto de dados desequilibrados, ou seja, quando existe uma classe com número de elementos destoantes do resto. A base de dados utilizada neste trabalho, como esperado, é desequilibrada, existem muito menos tweets classificados como “positivos” comparados com “neutros’ e “negativos”.

Em suma, o Complement NB estima-se probabilidades de características para cada classe y com base no complemento de y, ou seja, em todas as amostras de outras classes, em vez de apenas nas amostras de treinamento da própria classe.

Gaussian Naive Bayes

O classificador Gaussian Naive Bayes assume uma distribuição gaussiana para estimar a distribuição dos dados. Calcula-se as probabilidades de valores de entrada para cada classe usando uma frequência. Com entradas de valor real, calcula-se o desvio médio e padrão dos valores de entrada (x) para cada classe resumir a distribuição (Brownlee, 2016).

Em suma, além das probabilidades para cada classe, o Gaussian NB também armazena os desvios médios e padrão para cada variável de entrada em cada classe.

SVM – Support Vector Machines e Vizinhos Próximos

Para a implementação dos algoritmos classificadores SVM e Vizinhos próximos foram utilizados, respectivamente, os métodos: **Svm.SVC e KneighborsClassifier.**

2.9 AVALIAÇÃO DOS ALGORITMOS DE CLASSIFICAÇÃO

Com o objetivo de observar o comportamento e a acurácia dos algoritmos de classificação aplicados na base de dados, é implementado algumas métricas de validação da eficácia destes algoritmos, os seguintes métodos foram aplicados: *F-measure*, validação cruzada *k-fold* e método Predict.

O método F-measure é utilizado para situações em que se deseja ter apenas um resultado ao invés de dois para medir a performance. A pontuação do F-measure chega a 1 com um bom resultado e 0 com um resultado ruim, em suma, um valor alto de F-measure significa resultados de precisão e revocação balanceados. F-Measure é a média ponderada dos resultados de precisão e revocação. (Schreiber, Beskow, Muller, Nara, Silva e Reuter, 2017).

A validação cruzada k-fold é uma técnica computacional intensiva, que usa todas as amostras disponíveis como amostras de treinamento e teste, com isso consegue-se chegar a resultados mais preciso. Neste trabalho, foi definido k-fold=10, ou seja, a base de dados é dividida em 10 subconjuntos, após a divisão em subconjuntos, será utilizado um subconjunto na validação do modelo e os conjuntos restantes são utilizados como treinamento. (Schreiber, Beskow, Muller, Nara, Silva e Reuter, 2017).

O método de avaliação *Predict* é aplicado utilizando o split-test com 30% da base de dados para testes e 70% para treinamento, ou seja, diferente da validação cruzada, o método Predict não utiliza todas as amostras disponíveis como amostras de treinamento e testes.

3. DESENVOLVIMENTO

Neste capítulo, é apresentado todas as etapas do desenvolvimento do trabalho, desde o processo de extração dos tweets e criação da base dados, até o processo de aplicação dos algoritmos classificadores.

O capítulo 3 está organizado da seguinte forma:

- Sessão 3.1: Coleta de dados
- Sessão 3.2: Rotulação dos Sentimentos
- Sessão 3.3: Composição da base de dados
- Sessão 3.4: Nuvem de palavras
- Sessão 3.5: Pré-Processamento
- Sessão 3.6: Vetorização e Tokenizador
- Sessão 3.7: Implementação dos algoritmos de classificação

3.1 COLETA DE DADOS

A primeira etapa do desenvolvimento do trabalho é a coleta de dados, os dados em questão, são as mensagens publicada no Twitter, os *tweets*, e para fazer a extração do mesmo foi criado um *script* em *Python* utilizando a biblioteca tweepy. O script faz a extração automática dos tweets que se relacionam com as palavras-chaves definidas, ou seja, será retornado tweets em que no seu texto tenha a ocorrência da palavra-chave.

A extração dos tweets ocorre pelos seguintes passos:

3. Obtenção das *Keys e Tokens* da *API Twitter Standard*
4. Fazer a autenticação pela biblioteca *tweepy*
5. Implementar *Script* para extração dos *Tweets*

A Twitter API retorna um grande conjunto de dados com dezenas de informações de cada *tweet* capturado, como por exemplo: geolocalização, *time zone*, coordenadas, URL da imagem de perfil, nome do usuário e etc.

```
for tweet in tweets:
   print(tweet)
```

```
Status(_api=<tweepy.api.API object at 0x00000167BBE53190>, _json={'created_at': 'Tue Oct 27 21:34:50 +0000 2020', 'id': 13212
03729408937986, 'id_str': '1321203729408937986', 'full_text': 'Eu não tô reclamando de festa não, fui em várias durante a “pa
ndemia” Massssssssss não dá pra cair no papo de que estarão preocupados com saúde nos próximos 4 anos né kkkkkkkkkkkkkk', 't
runcated': False, 'display_text_range': [0, 182], 'entities': {'hashtags': [], 'symbols': [], 'user_mentions': [], 'urls':
[]}, 'metadata': {'iso_language_code': 'pt', 'result_type': 'recent'}, 'source': '<a href="http://twitter.com/download/iphon
e" rel="nofollow">Twitter for iPhone</a>', 'in_reply_to_status_id': None, 'in_reply_to_status_id_str': None, 'in_reply_to_use
r_id': None, 'in_reply_to_user_id_str': None, 'in_reply_to_screen_name': None, 'user': {'id': 772579626493370372, 'id_str':
'772579626493370372', 'name': 'Eduardo', 'screen_name': 'eduaf25', 'location': 'Minas Gerais, Brasil', 'description': 'o depo
is geralmente nem chega', 'url': None, 'entities': {'description': {'urls': []}}, 'protected': False, 'followers_count': 263,
'friends_count': 258, 'listed_count': 0, 'created_at': 'Sun Sep 04 23:38:31 +0000 2016', 'favourites_count': 2010, 'utc_offse
t': None, 'time_zone': None, 'geo_enabled': True, 'verified': False, 'statuses_count': 15940, 'lang': None, 'contributors_ena
bled': False, 'is_translator': False, 'is_translation_enabled': False, 'profile_background_color': '000000', 'profile_backgro
und_image_url': 'http://abs.twimg.com/images/themes/theme1/bg.png', 'profile_background_image_url_https': 'https://abs.twimg.
com/images/themes/theme1/bg.png', 'profile_background_tile': False, 'profile_image_url': 'http://pbs.twimg.com/profile_image
s/1320513772893671424/w4BuYXam_normal.jpg', 'profile_image_url_https': 'https://pbs.twimg.com/profile_images/1320513772893671
424/w4BuYXam_normal.jpg', 'profile_banner_url': 'https://pbs.twimg.com/profile_banners/772579626493370372/1598371750', 'profi
le_link_color': '001155', 'profile_sidebar_border_color': '000000', 'profile_sidebar_fill_color': '000000', 'profile_text_col
or': '000000', 'profile_use_background_image': False, 'has_extended_profile': False, 'default_profile': False, 'default_profi
le_image': False, 'following': None, 'follow_request_sent': None, 'notifications': None, 'translator_type': 'none'}, 'geo': N
```

Figura 13. Dados extraídos dos tweets

Como o objetivo do trabalho é capturar apenas a mensagem dos *tweets*, o *script* é configurado para retornar apenas o texto dos *tweets*.

```
for tweet in tweets:
   print(tweet.full_text)
```

```
Covid-19: Surgem mais evidências do papel da vitamina D ... Mais Revista Veja no post: https://t.co/PoEopoG0CL Muito mais infor
mação? https://t.co/WqoAnzSrL9 https://t.co/zquf3ZJvxf
Turismo mundial caiu 70% entre janeiro e agosto pela Covid-19 https://t.co/HE5zeeVBbl https://t.co/QMF7vFPTUd
a triste história de uma moça hidratada que só queria tomar um gole da água no metro porém covid-19
ala do covid-19 tomando uma injeção de boaz https://t.co/FrDJbxIUlh
Ceará registra mais de 271 mil casos e 9.323 óbitos por Covid-19 https://t.co/JFKX35pOfQ https://t.co/jVMIYhUcm6
eu lembro até hoje quando começou os primeiros casos de covid-19
tava andando com minhas amigas e falando, que isso não iria dar em nada, que seria só mais uma doença boba.
mano eu fico pensando, olha aonde nos chegou com essa merd@
@MBittencourtMD @hoc111 Esse deputado federal covidiota que chama máscara cirúrgica de "focinheira",em plena Covid-19,em um paí
s com quase 158 mil mortos pela doença, deveria responder por Crime contra a Saúde Pública e ,no Conselho de Ética por Quebra d
e Decoro Parlamentar.
Covid-19: Matosinhos encerra centros comerciais às 21:00 e pede medidas ao Governo https://t.co/r9kn69BqaP
Claudia Sheinbaum revela que dio positivo a COVID-19 https://t.co/pIz01yalpo https://t.co/LhZrLUAuOa
```

Figura 14. Exemplo de tweets apenas com o atributo texto

As palavras-chaves determinadas no *script* para a extração dos *tweets* e a quantidade total extraída são:

Palavras-chave para busca
#covid-19; covid-19; covid; coronavírus; corona vírus; corona; pandemia

Tabela 3. Palavras-chaves utilizadas na coleta de tweets

Total tweets capturados
1950 *tweets*

Tabela 4. Total de Tweets capturados

Após a captura, é utilizado a biblioteca *pandas* para criar um data frame com as colunas “*user*” e “*text*” e os *tweets* são armazenados em um arquivo CSV, que faz uma ordenação separando os valores com virgulas. Os tweets são posteriormente rotulados manualmente conforma o sentimento expresso no mesmo, entre “positivo”, “negativo” e “neutro”. O *script* de coleta foi configurado para extrair apenas *tweets* de língua portuguesa e ignorar *retweets*.

3.2 ROTULAÇÃO DOS SENTIMENTOS

Com 1950 tweets capturados e salvos em um arquivo CSV, é feito a classificação manual do sentimento expresso nos tweets, entre “positivo”, “negativo” ou “neutro”, o objetivo é obter o panorama total dos sentimentos expresso no Twitter, bem como, criar uma base para treino dos algoritmos classificadores.

A rotulação dos sentimentos entre “positivo”, “negativo” e “neutro” se deu devido aos seguintes critérios demonstrados na Tabela 5.

Positivo	**Negativo**	**Neutro**
Esperança	Raiva	Informações gerais
Otimismo	Pessimismo	Propaganda
Felicidade	Tristeza	Curiosidades
Sarcasmos positivo	Sarcasmos negativos	Conselho ou dicas
Ironias positivas	Ironias negativas	Ações governamentais
Recuperados da covid	Mortes	*Tweets* que não expressa sentimento
Desejar o bem e empenho contra a pandemia	Indignação e críticas	

Tabela 5. Critério de rotulação dos tweets

3.3 COMPOSIÇÃO DA BASE DE DADOS

Após o processo de rotulação de sentimentos dos tweets, é observado na Tabela 6 a composição final da base de dados, com o número total de tweets e de cada classe de sentimento.

Qtd. *Tweets*	1950
Positivo	340
Negativo	833
Neutro	777

Tabela 6. Configuração final da base

Gráfico de barras gerado com a configuração final da base (Figura 15). Como esperado, a maioria dos tweets sobre pandemia e o coronavírus contém sentimentos negativos e neutros, gerando uma base desbalanceada, com poucas amostras de classe positivo.

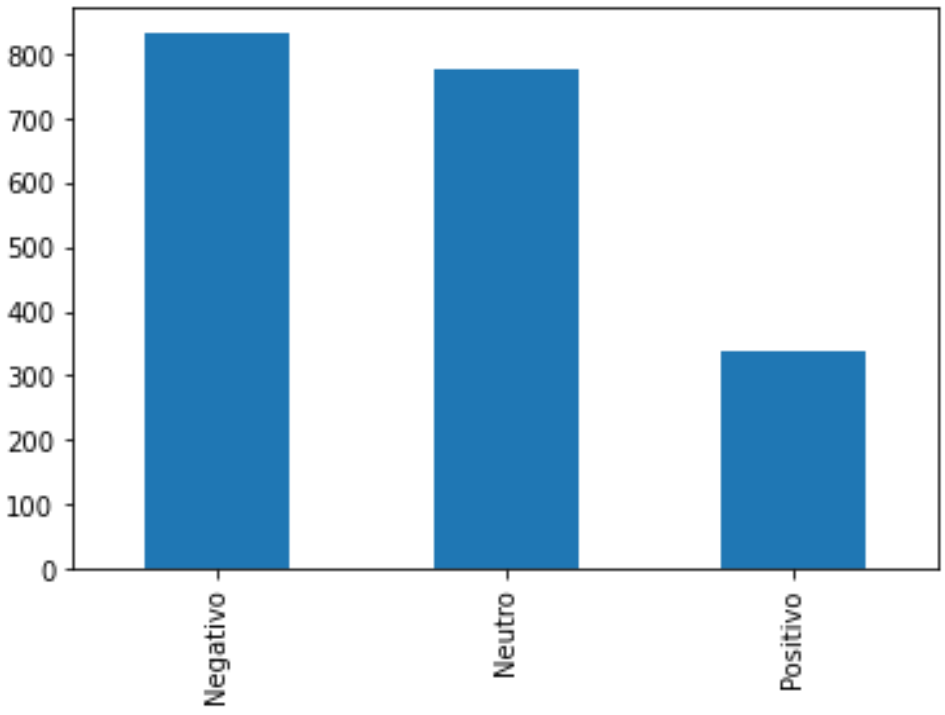

Figura 15. Gráfico de barras da base de dados

Exibição das 20 primeiras linhas da base de dados (Figura 16), em ordem de observar como estão dispostos os dados, bem como suas colunas e classificações.

```
dataset.head(20)
```

	Unnamed: 0	user	text	classe
0	0	iG	Governo do Amazonas afirma que mais de 300 mil...	Negativo
1	1	RdLitoral	A Jovem Pan News – Litoral conversou, na manhã...	Neutro
2	2	klebertorres	Qual a relação da covid-19 com a economia e in...	Neutro
3	3	Unicap	Alunos de Jornalismo e Psicologia desenvolvem ...	Neutro
4	4	CBNCuritiba	Pessoas que se recuperaram da Covid-19 podem d...	Positivo
5	5	studio877	#bentogonçalves #coronavírus | Bento Gonçalves...	Negativo
6	6	itsmarinareis	Eu não sei se eu to com sono/cansada ou com Co...	Negativo
7	7	CoiabAmazonia	"Intercâmbio de Experiências para a Proteção d...	Neutro
8	8	ClicRDC	Santa Maria é o bairro de Chapecó com mais cas...	Negativo
9	9	MarcosSantosAM	MPAM quer segurança sanitária contra Covid-19 ...	Neutro
10	10	Lusa_noticias	Covid-19: Primeiros 100 mil computadores distr...	Neutro
11	11	vborges91	@flavioamendola Único q pode estragar esse cli...	Negativo
12	12	ominhopt	Jogo da II Liga cancelado com os jogadores já ...	Neutro
13	13	JdLuxemburgo	Covid-19: Gil Vicente tem um jogador e três el...	Negativo
14	14	Jadhiel_aires	@LuizCamargoVlog O jornalista da UOL é tão des...	Positivo
15	15	patosnoticiasmg	Covid-19: porque idosos devem tomar mais cuida...	Neutro
16	16	Juristas	O evento acontece na próxima segunda-feira (14...	Neutro
17	17	biamora	Levar outra facada? Ter Covid-19 outra vez? Op...	Negativo
18	18	affonsoritter	Seis meses depois, pandemia poupa apenas 10 pa...	Negativo
19	19	_amotta	Quase 130 mil mortos, 4,23 milhões de infectad...	Negativo

Figura 16. Vinte primeiras linhas da base de dados

3.4 NUVEM DE PALAVRAS

Com os dados da coluna “text” da base de dados, contendo todos os tweets capturados e com a remoção de stopwords, é gerado uma nuvem de palavras, contendo uma representação visual da frequência de termos encontradas, mostrado proporcionalmente pelo tamanho da fonte (Figura 17).

Figura 17. Nuvem de palavras

3.5 PRÉ-PROCESSAMENTO

Nesta etapa é feito a limpeza dos dados, com o propósito de descartar o que é irrelevante para a etapa de classificação. As etapas de pré-processamento realizada na base de dados são as seguintes:

- **Remoção de stopwords:** Stopwords são artigos, preposições, pronomes e etc. São palavras que não possuem relevância para o resultado da classificação de texto.
- **Conversão de letras maiúsculas em minúsculas:** Com isso, é feito a padronização do texto.
- **Remoção de Links:** Links são termos que não possuem qualquer conteúdo semântico, logo não tem relevância para a classificação.
- **Remoção de RT – Retweets:** Para agilizar o processo de limpeza de dados, o script que faz a captura dos tweets pela Twitter API, foi configurado para não extrair retweets.
- **Remoção de caracteres não alfabéticos, como o &**: Novamente, o termo “&” não tem conteúdo semântico, portanto, é removido da base.
- **Remoção de linhas duplicadas, ou seja, tweets repetidos:** Nem todos os tweets repetidos são retweets, logo se fez necessário um script para excluir da base de dados os tweets repetidos, ou seja, será mantido apenas o primeiro encontrado.
- **Remoção de células vazias:** Para aplicar os algoritmos de classificação, os dados da base devem estar preenchido de forma que a o lado X (tweets) e o Y(sentimento) não fiquem desbalanceados, foi aplicado um script para excluir linhas da base de dados em que contenha alguma célula vazia.

Em ordem de demonstrar a transformação de um texto durante as fases de pré-processamento em um tweet existente na base de dados é produzido a *Tabela* **7** na próxima página.

Etapas da limpeza de dados	Tweet de exemplo e Resultados
Tweet Original	Vice-prefeito de Uberaba morre vítima da Covid-19 https://t.co/XgQmHMeDkl
Remoção de stopwords	Vice-prefeito Uberaba morre vítima Covid-19 https://t.co/XgQmHMeDkl
Conversão para letras minúsculas	vice-prefeito uberaba morre vítima covid-19 https://t.co/XgQmHMeDkl
Remoção de Links	vice-prefeito uberaba morre vítima covid-19
Aplicando todas as etapas acima + Remoção de caracteres não alfabéticos	
Tweet Original	Notícias & Atualizações do #Coronavirus - Perdeu algo? Ouça o #Podcast https://t.co/8NL8l4zczD
Aplicação de remoção de stopwords, links e conversão para minúsculas	notícias &amp atualizações #coronavirus perdeu algo? ouça #podcast
Remoção de caracteres não alfabéticos	notícias atualizações #coronavirus perdeu algo? ouça #podcast

Nota-se que apesar dos caracteres “&” e “#” serem ambos não alfabéticos, apenas o “&” foi removido, pois, no Twitter, as #*Hashtags* são importantes. Os métodos de pré-processamento escolhidos devem levar em consideração as características e o tipo da base de dados em que está sendo aplicado.

Aplicando a Remoção de células vazias e *tweets* repetidos	
Etapas	**Quantidade *tweets* restantes**
Número de *tweets* na base de dados	1951
Remoção de células vazias	1950
Remoção de *tweets* repetidos	1932

Tabela 7. Exemplo de aplicação das etapas de pré-processamento

3.6 VETORIZAÇÃO E TOKENIZADOR

Para a implementação dos algoritmos classificadores, é necessário que os dados estejam vetorizados. Neste trabalho foi utilizado o método CountVectorizer, sendo responsável pela vetorização dos dados de texto e a transforma-los em uma matriz de frequência, porém, para que os dados de textos sejam vetorizados, é preciso definir quais serão as palavras identificadas.

O Tokenizador é responsável por separar os termos(palavras) em tokens, sua função é identificar quais são as palavras no texto, Tokenizadores comuns identificam uma palavra a cada espaço encontrado, a escolha de um Tokenizador deve levar em consideração o conjunto de dados na qual ele será aplicado. A base de dados deste trabalho são 1950 tweets, e na rede social existem algumas peculiaridades de como os dados são apresentados, logo, foi aplicado um Tokenizador específico para dados do Twitter, o TweetTokenizer, disponível na biblioteca NLTK.

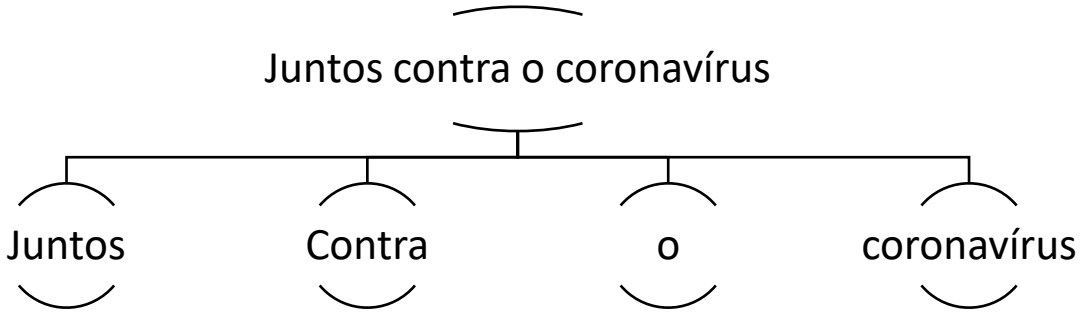

Figura 18. Tokenização

Na imagem acima podemos identificar como funciona um Tokenizador, ele quebra um texto em vários pedaços, separando as palavras usando espaços e pontuação.

No Twitter, os textos possuem peculiaridades que devem ser avaliadas, na tabela a seguir, dado um exemplo de tweet, como se comportaria um Tokenizador comum em comparação com o TweetTokenizer.

TWEET: @João Meu teste deu negativo #Alívio :) =D

Tokenizador Comum	**TweetTokenizer**
“@”,	“@João”,
“João”,	“Meu”,
“Meu”,	“teste”,
“teste”,	“deu”,
“deu”,	“negativo”,
“negativo”,	“#Alívio”,
“#”,	“:)”,
“Alívio”,	“=D”
“:”,	
“)”,	
“=”,	
“D”	

Tabela 8. Comparação de Tokenizadores

No comparativo dos Tokenizadores da Tabela 8, nota-se a diferença de comportamento entre os Tokenizadores, o Tokenizador comum entende que “@João” são duas palavras diferentes, logo, é separado entre “@” e “João”, o que não é adequado, e o mesmo se repete com “#Alívio” e os emoticons “:)” e “=D” não são reconhecidos e são separados caractere por caractere. O TweetTokenizer está pronto para lidar com este tipo de dado, ele é um Tokenizador específico para Twitter, logo, ele entende que @João é uma única palavra e que os emoticons e palavras seguidas de “#” são um termo único e são identificados de maneira correta.

3.7 IMPLEMENTAÇÃO DOS ALGORITMOS DE CLASSIFICAÇÃO

```
In [20]: tweets_vetorizados = vectorizer.fit_transform(text)
         type(tweets_vetorizados)

Out[20]: scipy.sparse.csr.csr_matrix
```

Figura 19. Vetorização – Matriz Esparsa

Os dados de texto (tweets) são vetorizados e é gerado uma matriz esparsa (Figura 19), usada para armazenar dados, contendo uma grande quantidade de elementos com valor zero como mostra a Figura 20.

```
In [23]: tweets_vetorizados.A

Out[23]: array([[0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

Figura 20. Array

A próxima etapa é a criação do modelo com diferentes algoritmos classificadores, são eles: Multinomial Naive Bayes, Bernoulli Naive Bayes, Complement Naive Bayes, Gaussian Naive Bayes, SVM – Support Vector Machines e KNN (N-Vizinhos próximos). O objetivo é observar a acurácia e o comportamento de cada modelo de classificação aplicado na base de dados contendo 1950 tweets rotulados com os sentimentos dos mesmos, entre “positivo”, “negativo” e neutro. Os modelos serão avaliados e comparados.

Aplicando o método train-test split, a base de dados é dividida em 30% para teste e 70% para treinamento (Figura 21), estimando-se a performance do algoritmo de aprendizagem de máquina pelo método “Predict”. Também é utilizado o método de validação cruzada k-fold, que utiliza todas as amostras disponíveis como amostras de treinamento e teste, sendo um dos métodos mais precisos de avaliação.

```
In [140]: X = vectorizer.fit_transform(dataset["text"])
          y = dataset["classe"]

In [118]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
          print(X_train.shape)
          print(X_test.shape)
          print(y_train.shape)
          print(y_test.shape)

          (1352, 8379)
          (580, 8379)
          (1352,)
          (580,)
```

Figura 21. Método Train-test split

4. RESULTADOS

Neste capítulo, é apresentado os resultados da aplicação de todos os modelos classificadores, bem como a tabela de *classification report* e a matriz de confusão. O capítulo está organizado da seguinte forma:

- Sessão 4.1 - Multinomial Naive Bayes: Método MultinominalNB ()
- Sessão 4.2 - Bernoulli Naive Bayes: Método BernoulliNB ()
- Sessão 4.3 - Complement Naive Bayes: Método ComplementNB ()
- Sessão 4.4 - Gaussian Naive Bayes: Método GaussianNB ()
- Sessão 4.5 – SVM Support Vector Machines: Método svm.SVC(C=1.0)
- Sessão 4.6 – KNN Vizinhos Próximos: Método KneighborsClassifier (N_Neighbors =10)
- Sessão 4.7 – Sumarização das acurácias, melhores e piores modelos.

4.1 MULTINOMIAL NAIVE BAYES

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 67,75%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 64,8%

Tabela 9. Acurácia Método Predict e Validação Cruzada - MultinominalNB

Na Tabela 9, é observado os valores de acurácia gerados pela implementação do modelo de Machine Learning, através do algoritmo MultinomialNB da biblioteca *Scikit-Learn*, pelo método Predict comum utilizando as amostras do 'train-test-split' e pelo método de validação cruzada de 10 partes em que utiliza toda as amostras.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.63	0.35	0.45	334
NEGATIVO	0.63	0.84	0.72	828
NEUTRO	0.68	0.57	0.62	770
ACURÁCIA			0.65	1932
MÉDIA MACRO	0.65	0.59	0.60	1932
MÉDIA COM PESO	0.65	0.65	0.63	1932

Tabela 10. Classification Report - MultinominalNB

Em sequência, é gerado a tabela de classificação (Tabela 10), na qual é observado as métricas de precisão, revocação e *F1-Score* para cada classe (Positivo, Negativo e Neutro), bem como a métrica de acurácia por validação cruzada. Neste cenário é observado a diferença de desempenho do algoritmo para com os *tweets* 'positivos' com uma *F1-Score* de 0.45, bem abaixo dos índices dos tweets 'negativos' e 'neutros'. O baixo índice de *F1-Score*, que representa a média ponderada entre os resultados de precisão e revocação, se deve ao fato de a base de dados ser desbalanceada, ou seja, contendo muito menos classificações de tweets positivos do que negativos e neutros.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	699	112	17	828
NEUTRO	283	437	50	770
POSITIVO	123	95	116	334
TUDO	1105	644	183	1932

Tabela 11. Matriz de Confusão - MultinominalNB

A matriz de confusão (Tabela 11) contém o número de classificações exatas do algoritmo para cada classe de sentimento, com as colunas ‘Real’ e ‘Predito’ é possível comparar os dados que existem na base de dados com os dados que efetivamente foram classificados pela implementação do modelo. É observado que dos 334 *tweets* rotulados como positivo na base de dados, apenas 116 foram classificados como positivo pelo algoritmo.

4.2 BERNOULLI NAIVE BAYES

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 66,89%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 62,73%

Tabela 12. Acurácia Método Predict e Validação Cruzada - BernoulliNB

Na Tabela 12, é observado os valores de acurácia gerados pela implementação do modelo de Machine Learning, através do algoritmo BernoulliNB da biblioteca *Scikit-Learn,* pelo método Predict comum utilizando as amostras do ‘*train-test-split*’ e pelo método de validação cruzada de 10 partes em que utiliza toda as amostras, obtendo resultados de acurácia inferiores em comparação ao modelo anterior, MultinomialNB.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.77	0.10	0.18	334
NEGATIVO	0.62	0.85	0.72	828
NEUTRO	0.63	0.61	0.62	770
ACURÁCIA			0.63	1932
MÉDIA MACRO	0.67	0.52	0.51	1932
MÉDIA COM PESO	0.65	0.63	0.59	1932

Tabela 13. Classification Report - BernoulliNB

Na tabela de classificação (Tabela 13), é observado novamente os índices de precisão, revocação e *F1-Score.* Nota-se o baixo índice de revocação de 0.10, no qual afeta o cálculo de F1-Score que representa a média harmônica entre revocação e precisão. O baixo índice de revocação significa que o algoritmo teve uma ‘cobertura’ baixa, ou seja, teve dificuldades de identificar tweets de classe positiva, sendo o BernoulliNB um dos modelos de pior performance para com os dados da classe desbalanceada, apesar de seu desempenho relativamente bom e semelhante aos outros algoritmos nas classes ‘negativo’ e ‘neutro’.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	707	120	1	828
NEUTRO	290	471	9	770
POSITIVO	146	154	34	334
TUDO	1143	745	44	1932

Tabela 14. Matriz de Confusão - BernoulliNB

Na Tabela 14, temos as classificações realizadas pelo algoritmo para cada classe de sentimento (Positivo, Negativo, Neutro) comparado ao número real de

rotulações que existe na base de dados. É observado a baixa performance do algoritmo na classificação dos tweets positivos, dos 334 tweets de rótulo positivo na base de dados, o modelo conseguiu classificar corretamente apenas 34.

4.3 COMPLEMENT NAIVE BAYES

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 68,96%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 65,73%

Tabela 15. Acurácia Método Predict e Validação Cruzada - ComplementNB

Na Tabela 15, é observado os valores de acurácia gerados pela implementação do algoritmo ComplementNB, com os resultados pelo método *Predict* (amostras do *train-test-split*) e por validação cruzada (todas as amostras). O modelo obteve o maior índice de acurácia geral, superando os resultados das outras implementações Naive Bayes, bem como supera a performance dos algoritmos SVM e KNN.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.50	0.60	0.54	334
NEGATIVO	0.68	0.81	0.74	828
NEUTRO	0.73	0.52	0.61	770
ACURÁCIA			0.66	1932
MÉDIA MACRO	0.64	0.64	0.63	1932
MÉDIA COM PESO	0.67	0.66	0.65	1932

Tabela 16. Classification Report - ComplementNB

Na Tabela 16, é observado o alto índice da *F1-Score* para tweets de classe ‘positivo’ (0.54) em comparação com os outros algoritmos. Esse fator contribuiu para que o modelo ComplementNB obtivesse o maior valor de acurácia geral.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	670	95	63	828
NEUTRO	229	398	143	770
POSITIVO	80	52	202	334
TUDO	979	545	408	1932

Tabela 17. Matriz de Confusão - ComplementNB

Corroborando com a métrica de *F1-Score*, na matriz de confusão (Tabela 17), é observado a melhor performance com os dados da classe desbalanceada, classificando corretamente 202 dos 334 tweets positivos da base de dados.

4.4 GAUSSIAN NAIVE BAYES

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 62,75%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 59,52%

Tabela 18. Acurácia Método Predict e Validação Cruzada - GaussianNB

Na Tabela 15, é observado os valores de acurácia gerados pela implementação do último algoritmo da variação Naive Bayes do trabalho, o algoritmo GaussianNB, com os resultados pelo método *Predict* (amostras do *train-test-split*) e por validação cruzada (todas as amostras). O modelo gerado pelo método GaussianNB obteve o pior desempenho comparado aos outros algoritmos da família Bayes implementados e o segundo pior desempenho geral, com uma acurácia de ~59,5% por validação cruzada e ~62,7% pelo método *Predict*.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.43	0.44	0.44	334
NEGATIVO	0.62	0.73	0.67	828
NEUTRO	0.64	0.52	0.57	770
ACURÁCIA			0.60	1932
MÉDIA MACRO	0.57	0.56	0.56	1932
MÉDIA COM PESO	0.60	0.60	0.59	1932

Tabela 19. Classification Report - GaussianNB

Na tabela 19, é observado os dados da tabela de classificação, novamente com os índices de precisão, revocação e *F1-Score*. O modelo GaussianNB obteve o segundo melhor índice da *F1-Score* para a classe ‘positivo’. Desta forma, se posiciona como o segundo melhor desempenho em relação aos dados balanceados, empatado com o modelo gerado pelo algoritmo MultinomialNB, porém, em relação as classificações de ‘Negativo’ e ‘Neutro’, o modelo gerado pelo algoritmo GaussianNB obteve resultados inferiores comparado aos outros modelos da família Bayes, o que levou a sua acurácia geral para ~60%.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	605	140	83	828
NEUTRO	256	397	114	770
POSITIVO	104	82	148	334
TUDO	968	619	345	1932

Tabela 20. Matriz de Confusão - GaussianNB

Na Tabela 20, é observado o número exato de classificações realizadas pelo modelo em comparação a quantidade real de amostras para cada classe da base de dados.

4.5 SVM - SUPPORT VECTOR MACHINES

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 64,65%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 61,85%

Tabela 21. Acurácia Método Predict e Validação Cruzada - SVM

Na Tabela 9, é observado os valores de acurácia gerados pela implementação do modelo SVM, através do algoritmo svm.SVC da biblioteca *Scikit-Learn*, pelo método Predict de validação cruzada. Desta forma, o modelo SVM obteve resultados superiores ao GaussianNB, porém, não performou melhor que as outras variações do Naive Bayes, mas manteve resultados próximos.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.86	0.19	0.31	334
NEGATIVO	0.62	0.76	0.69	828
NEUTRO	0.59	0.65	0.62	770
ACURÁCIA			0.62	1932
MÉDIA MACRO	0.69	0.53	0.54	1932
MÉDIA COM PESO	0.65	0.62	0.59	1932

Tabela 22. Classification Report - SVM

Na Tabela 22, observa-se novamente os índices de precisão, revocação e *F1-Score* do modelo. Com 0.19 de revocação na classe positiva, significa que a cobertura gerada pelo modelo SVM teve baixo desempenho, ou seja, o algoritmo não obteve um

bom desempenho na tarefa de identificar os tweets desta classe, o que contribuiu para o baixo índice final na *F1-Score*.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	633	194	1	828
NEUTRO	263	498	9	770
POSITIVO	120	150	64	334
TUDO	1016	842	74	1932

Tabela 23. Matriz de Confusão - SVM

Na matriz de confusão do modelo SVM (Tabela 23), nota-se o seu baixo desempenho com a classe desbalanceada, classificando apenas 64 tweets positivos. O modelo obtém desempenho melhores com as outras classes, porém, ainda um pouco abaixo das três primeiras implementações Naive Bayes nos índices de classificação da classe negativa, por exemplo.

4.6 KNN - VIZINHOS PRÓXIMOS (Kneighbors)

Avaliação: **Predição do modelo, método Predict – 30% teste e 70% treinamento**
Acurácia: 53,44%
Avaliação: **Método validação cruzada k-fold (10 partes)**
Acurácia: 53,26%

Tabela 24. Acurácia Método Predict e Validação Cruzada - KNN

Na Tabela 24, observa-se novamente os índices de precisão, revocação e *F1-Score* da implementação do algoritmo KNN - Vizinhos Próximos, que tem como método o *KneighborsClassifier* na biblioteca *Scikit-Learn*. Com uma acurácia de 53%, o modelo obteve a pior performance comparando-o com todas as outras implementações.

CLASSIFICATION REPORT

	PRECISÃO	***REVOCAÇÃO***	***F1-SCORE***	***QTD.AVALIADA***
POSITIVO	0.59	0.19	0.28	334
NEGATIVO	0.56	0.59	0.58	828
NEUTRO	0.50	0.62	0.56	770
ACURÁCIA			0.53	1932
MÉDIA MACRO	0.55	0.47	0.47	1932
MÉDIA COM PESO	0.54	0.53	0.52	1932

Tabela 25. Classification Report - KNN

Na Tabela 25, nota-se que o modelo gerado pela implementação do algoritmo KNN - Vizinhos Próximos obteve os piores índices de classificação em relação as classes ‘negativo’ e ‘neutro’ com *F1-Score* de 0.58 e 0.56 respectivamente, o modelo também obteve o segundo pior índice *F1-Score* para classe positiva com 0.28, superando apenas o modelo BernoulliNB na classe com dados desbalanceados.

MATRIZ DE CONFUSÃO

PREDITO	***NEGATIVO***	***NEUTRO***	***POSITIVO***	***TUDO***
REAL				
NEGATIVO	496	319	13	828
NEUTRO	265	481	24	770
POSITIVO	105	177	52	334
TUDO	866	977	89	1932

Tabela 26. Matriz de Confusão - KNN

Na matriz de confusão (Tabela 26), é observado novamente o número de classificações do modelo em comparação com a amostra total da base de dados em cada classe de sentimento.

4.7 S UMARIZAÇÃO DAS ACURÁCIAS

MÉTODO	AVALIAÇÃO PREDICT	VALIDAÇÃO CRUZADA
MULTINOMIAL NB	67,75%	64,80%
BERNOULLI NB	66,89%	62,73%
COMPLEMENT NB	68,96%	65,73%
GAUSSIAN NB	62,75%	59,52%
SVM	64,65%	61,85%
KNN	53,44%	53,26%

Tabela 27. Acurácia dos modelos

Na Tabela 27, é sumarizado os valores de acurácia de todos os algoritmos implementados, valores gerados pelo método Predict que utiliza as amostras definidas no método train-test-split e por validação cruzada de 10. Nas seguintes Tabelas (Tabela 28 e 29) temos os modelos que geraram a maior acurácia, sendo eles: ComplementNB e MultinomialNB, bem como os modelos que geraram os menores valores, sendo eles: KNN e GaussianNB.

MÉTODO	AVALIAÇÃO PREDICT	VALIDAÇÃO CRUZADA
COMPLEMENT NB	68,96%	65,73%
MULTINOMIAL NB	67,75%	64,80%

Tabela 28. Acurácia dos melhores modelos

MÉTODO	AVALIAÇÃO PREDICT	VALIDAÇÃO CRUZADA
KNN	53,44%	53,26%
GAUSSIAN NB	62,75%	59,52%

Tabela 29. Acurácia dos piores modelos

5. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Neste trabalho, foi apresentado de maneira breve, o surgimento dos meios digitais e o impacto das redes sociais para a ampliação de vozes e manifestações sociais, com isso, em um momento único em que o mundo atravessa a sua maior crise sanitária do século, as redes sociais se tornaram um ambiente propício para essas manifestações. O trabalho teve como objetivo promover uma análise de sentimentos das postagens na rede social Twitter sobre a pandemia da covid-19, bem como utilizar-se desse conjunto de dados capturados para promover uma avaliação e comparação de algoritmos classificadores de textos e processos de aprendizado de máquina.

O classificador Complement Naive Bayes, indicado para bases desbalanceadas, obteve a maior acurácia de classificação do modelo em relação às outras variantes do algoritmo Naive Bayes, bem como supera também os classificadores SVM e Vizinhos Próximos. A base de tweets, contém um baixo número de classificações de ‘tweets positivos’, por ser um classificador indicado para dados desbalanceados, o Complement NB obteve o melhor resultado na medida F-measure para tweets positivos (0.54), contribuindo para a média de acurácia geral do modelo.

O classificador Vizinhos Próximos, gerou o pior modelo de predição, não sendo de certa maneira surpreendente, devido ao mesmo não ser o mais indicado para classificação de texto. O Kneighbors Classifier é amplamente utilizado em técnicas para reconhecimento de faces.

O Multinomial Naive Bayes, é a variação Naive Bayes mais conhecida e amplamente utilizada para classificação de textos e análise de sentimentos, nesta aplicação, obteve resultados semelhantes ao Complement Naive Bayes, se posicionando como a 2º melhor acurácia geral. Em uma base de dados propriamente balanceada, a tendência é de que alcançaria o melhor resultado geral.

Na avaliação dos algoritmos, várias métricas e métodos diferentes foram utilizadas, são eles: os índices de Precisão e Revocação, na qual a F-measure é a média ponderada dos mesmos, o método Predict que avalia a acurácia do modelo de acordo com a base de teste e treinamento instanciada, e principalmente o método de

validação cruzada que resulta na acurácia utilizando toda o conjunto de dados como teste e treinamento, dividindo-os em subconjuntos, bem como é apresentado a tabela de *classification report* e a matriz de confusão, assim possibilitando uma ampla forma de avaliação dos modelos classificadores. Em relação a trabalhos futuros, existem algumas possibilidades:

- Promover uma comparação dos resultados obtidos com algoritmos de Machine Learning em relação a aplicação de Dicionários Léxicos de língua portuguesa.
- Aplicar métodos na programação em Pythom para remediar o problema da base desbalanceada e assim observando os novos resultados de acurácia.
- Utiliza-se de uma diferente ferramenta/método para extrair tweets em datas específicas e promover uma análise de sentimentos mais ampla e precisa em relação a covid-19

6. REFERÊNCIAS BIBLIOGRÁFICAS

ROCHA, E.; ALVES, L. Publicidade online: O pode das mídias e redes sociais (2010). - >http://revistas.pucgoias.edu.br/index.php/fragmentos/article/view/1371/917

WASSERMAN, S.; FAUST, K. *Social Network Analysis: Methods and Applications*, Pag. 20 (1994).

DEGENNE, A.; FORSE, M. *Introducing Social Networks* (1999).

RECUERO, R. Redes Sociais na Internet, Pag. 24 (2009)

GIRONÉS, L. *Geographical analysis of the opinion and influence of users during the coronavirus health crisis* (2020). - > https://riunet.upv.es/handle/10251/150975

FILHO, F.; COUTINHO, E. Uma análise de Tweets sobre coronavírus (2020). - > https://revistasmd.virtual.ufc.br/index.php/edicao-atual/#titulo3

CHAMOLA, V.; HASSIJA, V.; GUPTA, V.; GUIZANI, M. *A comprehensive Review of the COVID-19 Pandemic and the Role of IoT, Drones, AI, Blockchain, and 5G in Managing Its impact* (2020). - > https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9086010

CLEMENT, J. *Number of social networks users worldwide from 2017 to 2025 (2020), . Number of monthly active Twitter users worldwide from 1st quarter 2010 to first quarter 2019* (2020),

- > https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/ -

- > https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/

ALENCAR, V.; RODRIGUES, E.; MENDES, M.; PEIXOTO, S. Uso de Data Science Na Previsão de Febre Amarela Utilizando o Twitter, pag.2 (2019)

KOBLITZ, L. Ambiente de análise de sentimentos baseado em domínio (2010)

TOMAÉL, M.I.; ALCARÁ, A.R.; DI CHIARA, I.G. Das redes sociais à inovação (2005). - > https://www.scielo.br/pdf/ci/v34n2/28559.pdf/

DUARTE, F.; KLAUS, F. Redes Urbanas. In: DUARTE, F.; QUANDT, C.; SOUZA, Q. O Tempo Das Redes, p.156 (2008)

ARAUJO, G. Análise de Sentimento de mensagens do Twitter Em Português Brasileiro Relacionadas a Temas da Saúde (2014)

MARTELETO, M. Análise de redes sociais: aplicação nos estudos de transferência da informação (2001).

LOHSE, D. “Tuas ideias não correspondem aos fatos”: A ideologia da anti-ideologia no Twitter (2019).

LEITE, J. Mineração de textos do Twitter utilizando técnicas de classificação (2015).

CASTRO, J. Como funciona o Twitter? (2011). - > https://novaescola.org.br/conteudo/1957/como-funciona-o-twitter

WASHINGTON POST. *Twitter sees record numbers of users during pandemic, but advertising sales slow* (2020). - >

https://www.washingtonpost.com/business/economy/twitter-sees-record-number-of-users-during-pandemic-but-advertising-sales-slow/2020/04/30/747ef0fe-8ad8-11ea-9dfd-990f9dcc71fc_story.html

LIN, Y. 10 *Twitter statistics every marketer should know in 2020 [infographi*c] (2020). -

>https://www.oberlo.com/blog/twitterstatistics#:~:text=Here%27s%20a%20summary%20of%20the,are%20between%2035%20and%2065.

CASTRO, L.; FERRARI, D. Introdução á mineração de dados: conceitos básicos, algoritmos e aplicações – cap. 1.1 – 1.2(2016).

AMARAL, F. Aprenda mineração de dados: Teoria e prática, p. 2 (2016).

MARINHO, L.; GIRARDI, R. Mineração na Web, cap. 2 (?2003?).

SANTOS, R. Conceitos de Mineração de Dados na Web, p.18 cap. 2-3 (2009).

SANTOS, L. Protótipo para mineração de opinião em redes sociais: Estudo de casos selecionados usando o Twitter (2010).

CUNICO, F.; FOPPA, G. A mineração na Web, cap.2 (2016).

JUNIOR, J. Desenvolvimento de uma metodologia para mineração de Textos cap.2 (2008), L.;

GUIMARÃES, J. Elaboração e construção de um protótipo mínimo viável para o Tingoram: Um sistema de mineração de dados web baseado em georreferenciamento para sugestão semi automatizada de doação de alimentos, p. 17-18 (2018)

ARANHA, C.; PASSOS, E. A Tecnologia da Mineração de Textos (Artigo tutorial) (2006).

MORAIS, E.; AMBRÓSIO, A. Mineração de Textos – Instituto de Informática, Universidade Federal de Goiás (2007).

GONÇALVES, E. Mineração de Texto: Conceitos e Aplicações Práticas – SQL Magazine, v.105, 2012, p.31-44 (2012).

FIGUEIREDO, E.; CATINI, R.; MENDES, L. Mineração de Textos: Análise de Sentimento em Redes Sociais (2018) – Revisão Sistemática. Universidade Anhembi Morumbi-SP, FATEC-SP, UNIFACCAMP- SP. Anais da WCF, vol5. Pp 24-29, 2018, ISSN 2247-4703.

CHENG, Ching-Hsue. *A Text Mining Based on Refined Feature Selection to Predict Sentimental Review*. In: Proceedings of the Fifth International Conference on Network, Communication and Computing. ACM, 2016. p, 150-154

RODRIGUES, C.; VIEIRA, L.; MALAGOLI, L.; TIMMERMANN, N. Mineração de Opinião / Análise de Sentimentos. http://www.inf.ufsc.br/~luis.alvares/INE5644/MineracaoOpiniao.pdf

LIU, B. *Sentiment Analysis and Subjectivity. In. Handbook of Natural Language Processing*. 2nd., Toronto: Graeme Hirst, 2010.

MATIOLI, L. Protótipo para Mineração de Opiniões em Redes Sociais: Estudo de casos selecionados usando o Twitter. Monografia - Departamento de Ciências da Computação, Universidade Federal de Lavras, Minas Gerais, 2010.

CORRÊA, I. Análise dos sentimentos expressos na rede social Twitter em relação aos filmes indicados ao Oscar 2017 (2017). Universidade Federal de Uberlândia, Trabalho de conclusão de curso.

BRUNIALTI, L.; FREIRE, V.; PERES, S.; LIMA, C. Aprendizado de Máquina em Sistemas de Recomendação Baseados em Conteúdo Textual: Uma Revisão Sistemática (2015). XI Brazilian Symposium on Information System, Goiânia, May 26-29, 2015.

MONARD, M.; BARANAUSKAS, J. Conceitos sobre Aprendizado de Máquina (2003). Publicação – Sistemas Inteligentes cap.4.

GÉRON, A. Mãos á Obra: Aprendizado de Máquina com *Scikit-Learn & TensorFlow*: Conceitos, Ferramentas e Técnicas para a Construção de Sistemas Inteligentes (2019), Prefácio, Capítulo 1 e 2.

GOLDSCHIMDT, R.; PASSOES, E*. Data Mining*: um guia prático (2005) – Editora Campus, Rio de Janeiro.

SANTOS, F. Mineração de opinião em textos opinativos utilizando algoritmos de classificação. Monografia. Instituto de Ciências Exatas, Departamento de Ciência da Computação, Universidade de Brasília.

CARVALHO, L. *Data mining*: A mineração de dados no marketing, medicina, economia, engenharia e administração (2001). Editora Erica, 2001.

MITCHELL, T. *Machine Learning*. McGraw-Hill, 1997;

MARTINS, J. Classificação de páginas na internet. Trabalho de Conclusão (Mestrado). Instituto de Ciências Matemáticas e de Computação. USP. São Carlos, 2003.

ZHANG, H. *The optimality of Naïve Bayes* (2004). Proc. FLAIRS.

SCIKIT-LEARN. Naïve Bayes. - https://scikit-learn.org/stable/modules/naive_bayes.html

SINGH, G.; KUMAR, B.; GAUR, L.; TYAGI, *A. Comparison between Multinominal and Bernoulli Naïve Bayes for Text Classification* (2019). DOI: 10.1109/ICACTM.2019.8776800. *Conference: 2019 International Conference on Automation, Computational and Technology Management* (ICACTM).

GEEKS FOR GEEKS. *Complement Naïve Bayes (CNB) Algorithm* (2020) encurtador.com.br/itLQ9

BROWNLEE, J. *Naïve Bayes for Machine Learning (2016)*
https://machinelearningmastery.com/naive-bayes-for-machine-learning/

DORNELES, B. *Support Vector Machines* na identificação de opiniões depressivas em redes sociais. (2019) Universidade Federal da Grande Dourados, Trabalho de Conclusão de Curso, Curso de Engenharia de Computação.

SCHREIBER, J.; BESKOW, A.; MULLER, J.; NARA, E.; SILVA, J.; REUTER, J. Técnicas de validação de dados para sistemas inteligentes: Uma abordagem do software SDBAYES (2017). XVII Colóquio Internacional de Gestão Universitária, *Universidad Nacional de Mar del Plata*, Argentina. ISBN: 978-85-68618-03-5

CÂMARA, R. SVM – Entendendo Sua Matemática – Parte 1 – A Margem (2015)

http://www2.decom.ufop.br/imobilis/svm-entendendo-sua-matematica-parte-1-a-margem

PACHECO, A. K vizinhos mais próximos – KNN (2017).

http://computacaointeligente.com.br/algoritmos/k-vizinhos-mais-proximos/