

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Lucas Lima de Oliveira

**PREDIÇÃO DA POPULARIDADE DE TUÍTES UTILIZANDO  
ALGORITMOS DE APRENDIZADO DE MÁQUINA**

Santa Maria, RS  
2018

Lucas Lima de Oliveira

**PREDIÇÃO DA POPULARIDADE DE TUÍTES UTILIZANDO ALGORITMOS DE  
APRENDIZADO DE MÁQUINA**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Bacharel em Sistemas de Informação.**

ORIENTADOR: Prof. Sérgio Luís Sardi Mergen

Santa Maria, RS  
2018

**Lucas Lima de Oliveira**

**PREDIÇÃO DA POPULARIDADE DE TUÍTES UTILIZANDO ALGORITMOS DE  
APRENDIZADO DE MÁQUINA**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Bacharel em Sistemas de Informação**.

**Aprovado em 6 de dezembro de 2018:**

---

**Sérgio Luís Sardi Mergen, Dr. (UFSM)**  
(Presidente/Orientador)

---

**João Carlos Damasceno Lima, Dr. (UFSM)**

---

**Joaquim Assunção, Dr. (UFSM)**

Santa Maria, RS  
2018

## DEDICATÓRIA

*Dedico este trabalho à minha mãe, Siminea Lima, mulher guerreira e batalhadora que me ensinou a sempre correr atrás dos sonhos e superar os obstáculos. Cada ensinamento passado me permitiu trilhar esse caminho.*

## **AGRADECIMENTOS**

*Agradeço primeiramente a minha mãe, Siminea Lima e namorada, Amanda Rodrigues, por todo apoio e suporte prestado durante toda a jornada da graduação. Sem vocês ao meu lado, nada disse seria possível.*

*Agradeço também meu orientador, professor Sérgio Mergen, que além de ter participado da realização deste trabalho, também me deu suporte durante a graduação na realização de outras pesquisas. Agradeço imensamente por ter me acolhido nessa jornada, por todos os conselhos, amizade e por ter acreditado na minha capacidade.*

*Agradeço minha família e entes queridos, que sempre estiveram presentes e me apoiaram a cada passo dado. O apoio de todas essas pessoas foi fundamental nesta caminhada.*

*A todos os demais professores do Curso de Sistemas de Informação que sempre mostram dedicação e comprometimento em transmitir seus conhecimentos. Cada ensinamento será levado para toda a vida.*

*À Universidade Federal de Santa Maria, ao Centro de Tecnologia e à Coordenação do Curso de Sistemas de Informação por fornecer toda a estrutura necessária e proporcionar uma educação de qualidade.*

*Ao Programa de Educação Tutorial do curso de Sistemas de Informação e todos colegas que fizeram parte do grupo junto comigo. Todas as atividades realizadas pelo grupo me permitiram evoluir como profissional e como ser humano. Ter participado deste grupo foi uma honra e me proporcionou muitas oportunidades de aprendizado.*

*Agradeço também todos os colegas e amigos feitos durante essa trajetória que permitiram trocas de conhecimentos e experiências inesquecíveis. Espero poder levar essas amizades por toda a vida.*

*Que todos os nossos esforços estejam sempre focados no desafio à impossibilidade. Todas as grandes conquistas humanas vieram daquilo que parecia impossível.*

*(Charles Chaplin)*

## RESUMO

### PREDIÇÃO DA POPULARIDADE DE TUÍTES UTILIZANDO ALGORITMOS DE APRENDIZADO DE MÁQUINA

AUTOR: Lucas Lima de Oliveira

ORIENTADOR: Sérgio Luís Sardi Mergen

É conhecida a popularidade do Twitter e o poder que um único tuíte pode ter nos dias de hoje, servindo, inclusive, como fonte para portais de notícias renomados. A simplicidade e o volume de dados trafegados pela plataforma diariamente, fazem dela uma fonte de dados poderosa. Focando na análise das mensagens veiculadas e no interesse dos usuários em aumentar o número de seguidores, o objetivo deste trabalho é a elaboração de modelos, utilizando algoritmos de aprendizado de máquina, para realizar a predição e classificação da popularidade de tuítes com base em atributos extraídos do corpo das mensagens. Para alcançar esse objetivo, a metodologia adotada envolve a definição dos atributos de interesse, extração e processamento dos dados, além do estudo e aplicação de algoritmos de aprendizado de máquina para realizar a classificação dos tuítes.

**Palavras-chave:** Aprendizado de Máquina Supervisionado. Classificação de Dados. Coleta de Dados.

## **ABSTRACT**

### **PREDICTION OF TWEETS' POPULARITY USING MACHINE LEARNING ALGORITHMS**

AUTHOR: Lucas Lima de Oliveira  
ADVISOR: Sérgio Luís Sardi Mergen

The popularity of Twitter and the power that a single tweet can have today are well-known, even serving as a source for renowned news portals. The simplicity and volume of data trafficked by the platform daily make it a powerful data source. Focusing on the analysis of the messages and in the interest of the users to increase the number of followers, the objective of this work is the elaboration of models, using algorithms of machine learning, to make predictions and classifications of the tweets' popularity based on extracted attributes of the message body. In order to reach this objective, the methodology adopted involves the definition of interest attributes, extraction and processing of data, as well as the study and application of machine learning algorithms to perform the classification of tweets.

**Keywords:** Supervised Machine Learning. Data Classification. Data collection.



## LISTA DE FIGURAS

Figura 2.1 – Exemplo de Problemas de Classificação e Regressão. ....	15
Figura 2.2 – Exemplo da diferença entre as diferentes abordagens. ....	16
Figura 2.3 – Engenharia de Atributos no processo de Aprendizado de Máquina ..	17
Figura 2.4 – Exemplo de árvore para classificação de um tuíte como popular. ...	21
Figura 3.1 – Arquitetura adotada para extração de tuítes .....	25
Figura 4.1 – Variação no número de instâncias balanceadas por taxa de engajamento. ....	33
Figura 4.2 – Acurácia de cada algoritmo aplicado sobre toda a base com diferentes taxas de engajamento. ....	34
Figura 4.3 – Sensibilidade de cada algoritmo aplicado sobre toda a base com diferentes taxas de engajamento. ....	35
Figura 4.4 – Variação no número de instâncias em relação às métricas para o algoritmo Naive Bayes utilizando o texto. ....	35

## LISTA DE TABELAS

Tabela 2.1 – Exemplo de tabela com frequências de palavras e suas classes.....	20
Tabela 3.1 – Dados coletados para cada tuíte.....	26
Tabela 3.2 – Dados obtidos na etapa de Extração.....	28

## LISTA DE ABREVIATURAS E SIGLAS

<i>API</i>	<i>Application Programming Interface</i>
<i>ARFF</i>	<i>Attribute-Relation File Format</i>
<i>CART</i>	<i>Classification and Regression Trees</i>
<i>IA</i>	Inteligência Artificial
<i>LMT</i>	<i>Logistic Model Trees</i>
<i>LSTM</i>	<i>Long Short-Term Memory</i>
<i>PCA</i>	<i>Principal Component Analysis</i>
<i>RNN</i>	<i>Recurrent Neural Network</i>
<i>SVM</i>	<i>Support Vector Machine</i>
<i>WEKA</i>	<i>Waikato Environment for Knowledge Analysis</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>11</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>14</b>
2.1	APRENDIZADO DE MÁQUINA .....	14
2.1.1	<b>Aprendizado de Máquina Supervisionado .....</b>	<b>14</b>
2.1.2	<b>Aprendizado de Máquina Não Supervisionado .....</b>	<b>15</b>
2.2	ENGENHARIA DE ATRIBUTOS .....	16
2.3	ALGORITMOS DE APRENDIZADO DE MÁQUINA SUPERVISIONADO .....	18
2.3.1	<b>Naive Bayes .....</b>	<b>18</b>
2.3.2	<b>Árvores de Decisão .....</b>	<b>20</b>
<b>3</b>	<b>PROPOSTA .....</b>	<b>23</b>
3.1	DEFINIÇÃO DOS ATRIBUTOS DE INTERESSE .....	23
3.2	DEFINIÇÃO DE POPULARIDADE .....	24
3.3	PROCESSAMENTO DOS TUÍTES .....	25
3.3.1	<b>Coleta dos Tuítes .....</b>	<b>26</b>
3.3.2	<b>Extração dos Atributos .....</b>	<b>27</b>
3.3.3	<b>Atualização dos Dados de Retuítes e Curtidas .....</b>	<b>28</b>
3.4	CLASSIFICAÇÃO DOS TUÍTES .....	28
3.4.1	<b>Balanceamento das instâncias .....</b>	<b>29</b>
3.4.2	<b>Classificação utilizando Naive Bayes e Árvores de Decisão .....</b>	<b>30</b>
<b>4</b>	<b>EXPERIMENTOS .....</b>	<b>32</b>
4.1	MÉTRICAS DE AVALIAÇÃO .....	32
4.2	APLICAÇÃO DOS CLASSIFICADORES .....	33
<b>5</b>	<b>CONCLUSÕES .....</b>	<b>36</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>37</b>

# 1 INTRODUÇÃO

Com a grande popularização dos chamados influenciadores digitais, é notável o crescimento das mídias sociais como meios de comunicação e divulgação de conteúdos. Neste cenário, onde o número de seguidores determina a sua influência, torna-se muito importante que essas personalidades compreendam seu público, pois conteúdos direcionados refletem diretamente no alcance de suas publicações. Dentre as redes sociais mais utilizadas atualmente, o Twitter é um meio de veiculação de mensagens que se destaca, por sua simplicidade e objetividade. Embora não tenha o mesmo destaque que outras plataformas, como o Facebook ou o Instagram, o Twitter conta com cerca de 335 milhões de usuários ativos, segundo Statista <sup>1</sup>, e em média 500 milhões de tuítes que são publicados diariamente, segundo *Internet Live Stats* <sup>2</sup>, o que faz dessa rede uma fonte de dados muito poderosa.

Uma das preocupações de usuários do Twitter é alavancar sua popularidade, através do aumento no número de seguidores. Essa preocupação é fundamental para empresas e personalidades públicas que utilizam suas imagens para fins monetários. Nesses casos, o uso das redes sociais deve ser planejado e monitorado. Quando isso é realizado da maneira correta, a marca e/ou a pessoa ficam muito mais próximos de seus fãs e seguidores, o que conseqüentemente, faz sua popularidade e influência aumentar. Um dos indicadores capaz de medir a influência de um usuário em redes sociais é a taxa de engajamento, que leva em consideração as interações dos usuários com as publicações de uma página, dentre essas interações, podem ser considerados os retuítes, curtidas e comentários. Considerando esse fator, pode-se afirmar empiricamente que o aumento na quantidade de retuítes leva a um aumento na quantidade de seguidores, devido a propagação exponencial daquele conteúdo.

Tendo em vista o interesse dos usuários em aumentar o alcance de suas postagens, poder identificar os fatores que têm maior influência sobre a popularidade de suas mensagens pode ser uma grande vantagem ao tentar aumentar o engajamento por parte de seus seguidores. Ser capaz de prever/estimar a popularidade que um tuíte poderá obter, baseando-se nas características presentes no corpo de sua mensagem, permite a realização de diferentes análises a cerca o conteúdo disseminado por aquela conta, o que pode trazer muitos benefícios aos usuários com relativa influência nessa rede social.

Como afirma (SUH et al., 2010), a propagação de um tuíte está diretamente ligada ao conteúdo e valor informativo contido nele. Nesse sentido, os autores avaliaram um conjunto de características extraídas das mensagens. Os resultados mostra-

---

<sup>1</sup> Statista: <https://www.statista.com/topics/737/twitter/>

<sup>2</sup> Internet Live Stats: <http://www.internetlivestats.com/twitter-statistics/>

ram que a utilização de *hashtags* e URLs são fatores muito significativos e que ajudam a impulsionar uma publicação. Apesar de ser um resultado muito relevante, o trabalho não realizou uma análise exaustiva das características que podem ser extraídas do corpo das mensagens de cada tuíte.

É conhecido que hoje existem inúmeras pesquisas sendo realizadas envolvendo dados coletados do Twitter. Além de (SUH et al., 2010), outros trabalhos relacionados que podem ser citados aqui, como (DUAN et al., 2010), (BENEVENUTO et al., 2010), (NAVEED et al., 2011), (KHARDE; SONAWANE, 2016) e (XU; YANG, 2012), tem como parte de seus objetivos, a análise e identificação de fatores impactantes no conteúdo das mensagens propagadas no Twitter, além disso, utilizam também técnicas de aprendizado de máquina na realização de suas pesquisas.

Dentre as principais técnicas utilizadas nestes trabalhos, estão: Máquina de Vetores de Suporte (SVM, do inglês: *Support Vector Machine*); árvores de decisão, que em alguns dos casos são utilizadas para identificar a importância e selecionar os atributos; Naive Bayes; e Regressão logística. No caso dos atributos utilizados nestas pesquisas, foram considerados como fatores de relevância: utilização de URLs e *hashtags*; o alcance do autor do tuíte (podendo ser medido por seus seguidores ou métricas mais complexas); sentimento da mensagem; além dos fatores de popularidade relacionados a cada tuíte, como as curtidas e os retuítes.

Ainda que cada um destes trabalhos apresentem contribuições muito significativas no contexto da descoberta de conhecimento através de dados coletados do Twitter com aprendizagem de máquina, as análises e experimentos são realizados de maneira genérica, sendo aplicadas as mesmas regras para todos os tipos de usuários. Além disso, as análises sobre os modelos não medem a qualidade das previsões com base em fatores variáveis de engajamento.

Dentro deste contexto, o objetivo deste trabalho é elaborar modelos, utilizando algoritmos de aprendizado de máquina, para realizar a predição e classificação da popularidade de tuítes, com base em suas características e considerando uma taxa de engajamento variável. Para formar a base de dados, também é estabelecido como parte do objetivo monitorar e extrair tuítes de determinadas contas do Twitter que possuam certo grau de influência.

No que se refere a aprendizagem de máquina, serão estudados e testados algoritmos já consolidados, como Naive Bayes e árvores de decisão, como entrada para estes algoritmos, serão utilizados dados provenientes do pré-processamento dos tuítes coletados, sendo consideradas as seguintes características: o tamanho em caracteres; o sentimento (que mede a emoção transmitida); a banalidade (que mede a relevância da mensagem); a presença de *hashtags* e URLs, que também foram utilizados nos trabalhos relacionados, além do próprio texto do tuíte. A escolha dos algoritmos de classificação foi feita com base em sua popularidade e destaque para

solucionar o problema de classificação de dados.

Este trabalho está estruturado nas seguintes seções. O capítulo 2 apresenta a fundamentação teórica, abordando conceitos e algoritmos de aprendizado de máquina. O capítulo 3 apresenta a definição dos atributos e a arquitetura de extração de tweets usada, que realiza desde a coleta até a preparação dos dados para análise. O capítulo de Experimentos apresentará as análises realizadas a partir dos dados coletados juntamente com a aplicação dos algoritmos de aprendizado de máquina estudados. O capítulo de Conclusões apresenta as considerações finais a cerca do trabalho realizado.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os conceitos relacionados ao aprendizado de máquina, na seção 2.1, definindo as diferenças entre a aprendizagem supervisionada e a não supervisionada. Em seguida, na seção 2.3, são apresentados alguns dos principais algoritmos do segmento supervisionado, os quais também foram utilizados na realização de experimentos no decorrer deste trabalho, sendo estes o Naive Bayes e algoritmos populares para a indução de Árvores de Decisão.

### 2.1 APRENDIZADO DE MÁQUINA

Entende-se como sistemas inteligentes, aqueles que são capazes de processar dados de entrada e ajustar padrões internos a fim de otimizar seus resultados de saída, de acordo com os objetivos esperados para aquele algoritmo. Dentro deste contexto, o aprendizado de máquina foca no treinamento desses algoritmos para melhorar seu desempenho. Esse processo está ligado com a redução de dimensionalidade, classificação e associação dos dados e previsão de comportamentos.

Algoritmos de aprendizado de máquina (ou *machine learning* em inglês) dividem-se em dois segmentos, aqueles que necessitam de uma supervisão para melhorar seus resultados e aqueles fazem esse processo de maneira independente. Nesta seção serão apresentados esses dois tipos de algoritmos, especificando suas características e diferenças.

#### 2.1.1 Aprendizado de Máquina Supervisionado

A aprendizagem supervisionada realiza o treinamento dos algoritmos com dados para os quais suas respostas já sejam conhecidas. Ou seja, dependem sempre da entrada de um padrão de valores e da comparação das respostas do sistema com aquelas consideradas corretas. Conforme o algoritmo é treinado seus padrões vão sendo ajustados a fim de diminuir o erro e otimizar as respostas.

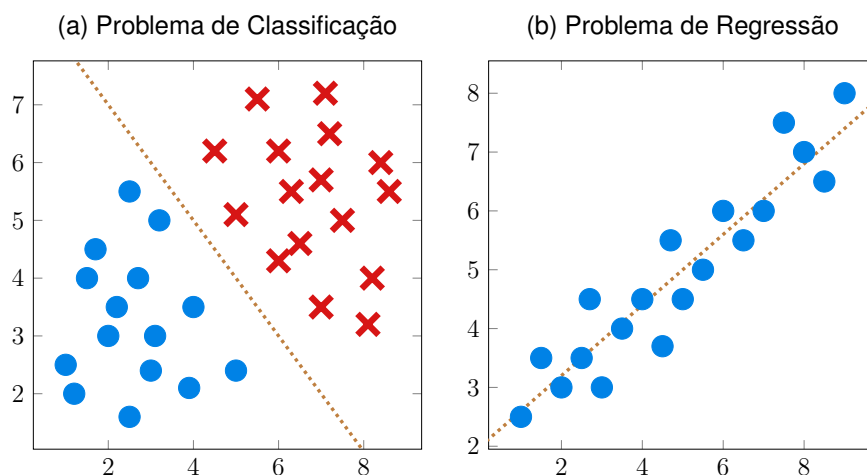
Os problemas solucionados através da aprendizagem supervisionada são divididos em problemas de regressão e classificação de dados, como ilustra a Figura 2.1. Segundo (RUSSELL; NORVIG, 2014), quando o resultado esperado pelo algoritmo for um conjunto finito de valores, (como fraco, mediano ou forte), trata-se de problema de classificação, pois os dados de entrada devem ser categorizados dentro daquele



grupo. No caso do resultado esperado ser numérico, trata-se de um problema de regressão, na qual tenta-se identificar uma tendência nos valores com base nos dados de entrada.

Nesse tipo de aprendizagem o algoritmo recebe as entradas já categorizadas para realizar o treinamento e, a cada iteração, ajusta seus parâmetros para obter a melhor saída, podendo ser, por exemplo, minimizar o erro, maximizar a precisão ou a acurácia. Frequentemente, após a etapa de treinamento, é realizada uma etapa de validação, passando ao algoritmo entradas sem classificação, dessa forma seu desempenho pode ser realmente avaliado e, se necessário, o treinamento pode ser realizado novamente com novos ajustes em seus parâmetros.

Figura 2.1 – Exemplo de Problemas de Classificação e Regressão.



Fonte: Produção do próprio autor.

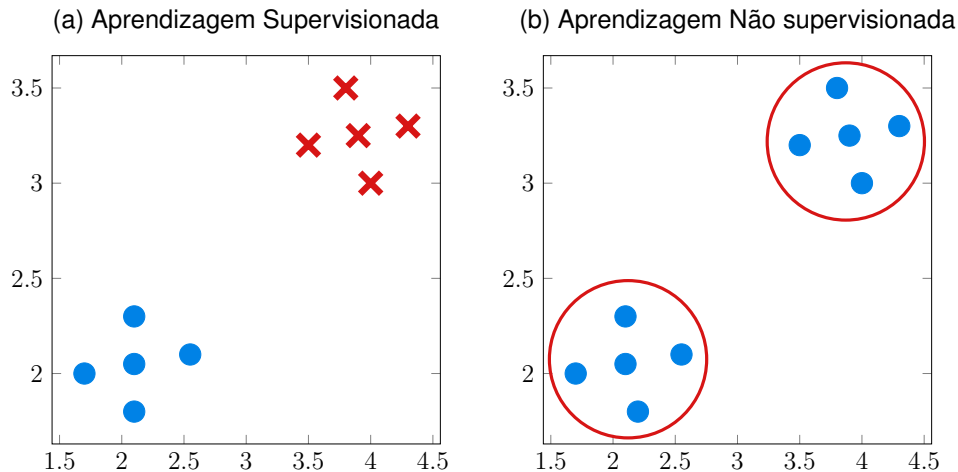
### 2.1.2 Aprendizado de Máquina Não Supervisionado

No caso dos algoritmos de aprendizagem não supervisionada, ao contrário do segmento apresentado na subseção anterior, estes recebem os dados sem nenhuma classificação prévia, impossibilitando o aferimento das classes de cada entrada. Consequentemente, conforme os dados vão sendo recebidos, o próprio algoritmo é responsável por identificar as relações e padrões presentes nos dados, o que por si só pode ser considerado um objetivo a ser alcançado. A aprendizagem não supervisionada não prevê soluções específicas para realizar o treinamento e validação dos resultados, ou seja, não há um *feedback* explícito sobre os resultados previstos.

Como explica (RUSSELL; NORVIG, 2014), o exemplo mais comum de aprendizagem não supervisionada, é o de agrupamento, onde o objetivo é detectar grupos potencialmente úteis dentro dos valores de entrada, que podem ser semelhantes ou

estar relacionados por diferentes variáveis. A Figura 2.2 exemplifica as diferenças entre esses dois tipos de abordagem.

Figura 2.2 – Exemplo da diferença entre as diferentes abordagens.



Fonte: Produção do próprio autor.

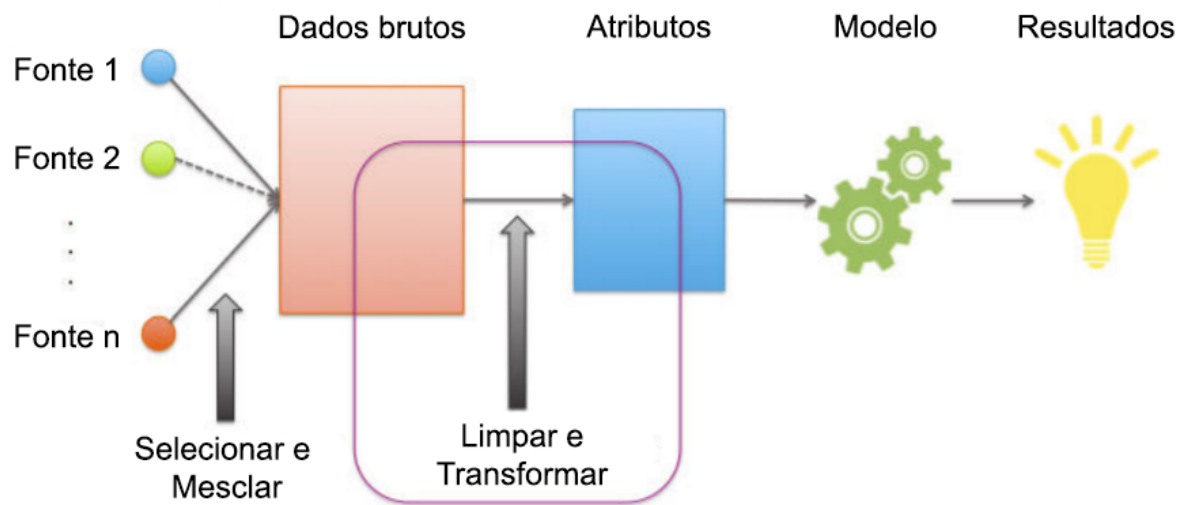
## 2.2 ENGENHARIA DE ATRIBUTOS

Engenharia de atributos (ou *Feature Engineering*) é um conjunto de técnicas muito utilizadas no processo de aprendizado de máquina. Essas técnicas tem o objetivo de agregar valor mais significativo aos dados coletados e assim, melhorar os modelos de IA (ZHENG; CASARI, 2018). Muitas vezes, o processo de engenharia de atributos por si só pode trazer melhores resultados independente dos algoritmos utilizados. Como já disse Peter Norvig, escritor de renomados livros na área de inteligência artificial e atualmente diretor de pesquisa no Google, "mais dados batem algoritmos inteligentes, mas dados melhores batem mais dados", ou seja mesmo utilizando algoritmos poderosos de aprendizagem de máquina, os resultados não serão tão bons se os dados não forem significativos.

A escolha de quais técnicas utilizar está ligada, na maioria dos casos, tanto aos dados quanto ao modelo, pois alguns deles podem estar mais adaptados a um determinado tipo de atributo (ZHENG; CASARI, 2018). A Figura 2.3 mostra onde a aplicação de engenharia de atributos se localiza no processo de aprendizagem de máquina e torna claro que as escolhas tomadas em qualquer uma das etapas tem influência direta nas suas subsequentes.

Como mencionado, existem inúmeras técnicas de engenharia de atributos dependendo do tipo de dados, e para cada uma delas, existem diferentes estratégias para sua aplicação. Para dados numéricos, podem ser aplicadas técnicas como: pre-

Figura 2.3 – Engenharia de Atributos no processo de Aprendizado de Máquina



Fonte: Traduzido de (ZHENG; CASARI, 2018)

enchimento de valores faltantes, que tem por objetivo evitar a perda de informações; arredondamento de valores, pois muitas casas decimais podem representar ruídos; transformação logarítmica, para reduzir a diferença na escala entre valores muito grandes e outros muito pequenos; normalização, para padronizar a escala dos valores; dentre outras.

No caso de dados textuais, existem técnicas para realizar a preparação do texto, ou pré-processamento, que podem envolver alguns procedimentos como transformação em letras minúsculas, lematização das palavras, remoção de acentuações, caracteres não textuais e palavras comuns (ou *stopwords*). Além disso, outros métodos de engenharia de atributos podem ser aplicados, como vetorização do texto com técnicas como *bag-of-words* e *word embeddings* (respectivamente bolsa de palavras e incorporação de palavras, em tradução livre), que buscam determinar a frequência e similaridade das palavras presentes nos textos.

Em consequência de todos os procedimentos, muitos atributos novos podem ser gerados e o processamento de todos eles pode se tornar muito custoso. Nestes casos, existem abordagens com o intuito de selecionar os atributos mais representativos dentre todos disponíveis. Dentre estas abordagens de seleção, pode-se citar Análise de Componentes Principais (PCA, do inglês: *Principal Component Analysis*); método de filtragem, que busca identificar correlações entre os dados; método *Wrapper* (ou de embrulhar, em tradução livre), que é baseado na tentativa e erro para encontrar a melhor combinação de atributos; e o método *Embedded* (ou incorporado, em tradução livre), casos em que a seleção destes atributos já faz parte do modelo.

## 2.3 ALGORITMOS DE APRENDIZADO DE MÁQUINA SUPERVISIONADO

Dentro do escopo deste trabalho, que tem como um dos objetivos realizar a predição da popularidade de tuítes, requer-se a utilização da aprendizagem de máquina supervisionada, pois os resultados esperados estão diretamente ligados a classificação dos dados. Como mencionado, nesta seção serão abordados alguns dos principais algoritmos que se encaixam neste segmento e que serão utilizados no decorrer deste trabalho, apresentado suas características, funcionamento, vantagens e desvantagens na sua utilização.

### 2.3.1 Naive Bayes

A técnica Naive Bayes pode ser considerada como uma das mais populares para classificação de dados utilizando aprendizado de máquina. O algoritmo utiliza de métodos probabilísticos, baseados na Teoria Bayesiana, criada por Thomas Bayes no século XVIII. Para compreender melhor o funcionamento dessa técnica, é importante entender também um pouco sobre o teorema do qual ela teve origem.

Como mostra (RUSSELL; NORVIG, 2014), o teorema, ou regra de Bayes é uma formula simples, definida pela Equação 2.1, que vem da regra do produto de probabilidades, assumindo que  $prob(D|H) = prob(H|D)$ , sendo  $H$  a hipótese a ser validada e  $D$  os dados observados, podendo ser tratados também como *causa* e *efeito*. Apesar de simples, essa regra é a base de grande parte dos sistemas de IA (Inteligência Artificial) que utilizam inferência probabilística.

$$prob(H|D) = \frac{prob(D|H)prob(H)}{prob(D)} \quad (2.1)$$

Dividindo as partes do teorema, do lado esquerdo,  $prob(H|D)$  é chamada de probabilidade posterior da hipótese após a realização do experimento; do lado direito,  $prob(D|H)$  chamada função de verossimilhança, é a distribuição de probabilidade dos dados, a qual multiplica-se por  $prob(H)$ , denominada *Prior*, que é a probabilidade da hipótese ser verdadeira; por fim, o denominador  $prob(D)$ , é a probabilidade total.

Ainda que possa parecer um teorema simples, seu alcance está na sua capacidade de interpretação. No caso do modelo Naive Bayes, ou Bayes Ingênuo, assume-se que os atributos *efeito* são condicionalmente independentes entre si, dada a *causa* – daí a denominação de “ingênuo”. A distribuição probabilística deste modelo pode ser descrita conforme indica a Equação 2.2, sendo  $C$  a classe, ou causa, que deve ser prevista, enquanto que o conjunto  $\{x_1, \dots, x_n\}$  são os atributos, ou efeitos.

$$P(C|x_1, \dots, x_n) = \alpha P(C) \prod_i P(x_i|C) \quad (2.2)$$

Este modelo de aprendizagem é facilmente escalável para problemas maiores, funcionando muito bem com uma ampla variedade de aplicações, apesar de se destacar e ser comumente utilizado em uma série algoritmos para classificação de textos. Além disso, este modelo não apresenta grandes complicações com dados ruidosos ou faltantes, podendo inclusive realizar previsões adequadas nestes casos. Esses fatores fazem o Naive Bayes ser (provavelmente) o modelo de rede Bayesiana mais comumente utilizado em algoritmos de aprendizado de máquina.

Tomando como exemplo a clássica classificação de sentimentos em textos, como mencionado, o algoritmo irá assumir que as palavras de uma determinada mensagem não possuem uma relação entre si. Sendo assim, o classificador poderá presumir que uma frase seja positiva, caso a maioria das palavras presentes nela tenham maior probabilidade de ter este mesmo sentimento, independentemente do contexto em que foram utilizadas.

Para classificar uma determinada frase, inicialmente é preciso montar uma base de treinamento, contendo a classificação dos dados de entrada, que no caso da análise de sentimentos, será positivo ou negativo. A partir destes dados, é criada uma tabela para guardar a frequência de cada uma das entradas com suas classes e a probabilidades de cada entrada. Para testar uma nova entrada, é calculada sua probabilidade para cada uma das possíveis classificações com base nas ocorrências anteriores. Para os casos em que o dado de teste não está presente na base de treinamento ou não foi classificado para uma das classes, existem algumas técnicas capazes de corrigir esse problema. Uma técnica muito comum aplicada para estes casos é a suavização de Laplace, a qual soma o valor 1 para todos os valores, desta forma, nenhuma operação é realizada utilizando o valor 0.

Utilizando como exemplo a frase “*With great power comes great responsibility*”, e considerando a Tabela 2.1 (fictícia) apresentada logo abaixo, na qual consta a frequência das palavras para cada classe e a probabilidade de cada uma, para classificar a palavra “*great*” como popular ou impopular, considerando também que essa possa ser uma frase extraída do Twitter, seriam realizadas as seguintes operações listadas a seguir.

Tabela 2.1 – Exemplo de tabela com frequências de palavras e suas classes.

Palavras	Popular	Impopular	Probabilidade
responsibility	1	2	3/14 = 0,21
power	2	1	3/14 = 0,21
great	3	1	4/14 = 0,28
bad	0	2	2/14 = 0,14
good	2	0	2/14 = 0,14
Total	8	6	
		Positivo	8/14 = 0,57
		Negativo	6/14 = 0,42

Fonte: Produção do próprio Autor.

$$P(\text{great}|\text{popular}) = 3/8 = 0.37 \quad (2.3)$$

$$P(\text{popular}) = 8/14 = 0.57 \quad (2.4)$$

$$P(\text{great}) = 4/14 = 0.28 \quad (2.5)$$

$$P(\text{great}|\text{unpopular}) = 1/6 = 0.16 \quad (2.6)$$

$$P(\text{unpopular}) = 6/14 = 0.42 \quad (2.7)$$

$$P(\text{popular}|\text{great}) = 0.37 * 0.57 / 0.28 = 0.75 \quad (2.8)$$

$$P(\text{unpopular}|\text{great}) = 0.16 * 0.42 / 0.28 = 0.24 \quad (2.9)$$

A partir dos cálculos realizados, com base na Tabela 2.1 apresentada, obtém-se como resultado uma probabilidade maior para a palavra ‘great’ ser popular. Para realizar a classificação considerando toda a frase, essa operação é aplicada para cada palavra, as probabilidades resultantes para cada classe são multiplicadas e os resultados são aplicados na regra de Bayes, conforme a Equação 2.1, para cada uma das possíveis classes. Mesmo sendo um exemplo simples da aplicação da técnica Naive Bayes, é possível observar a facilidade da aplicação deste algoritmo para a classificação de dados utilizando um método probabilístico.

### 2.3.2 Árvores de Decisão

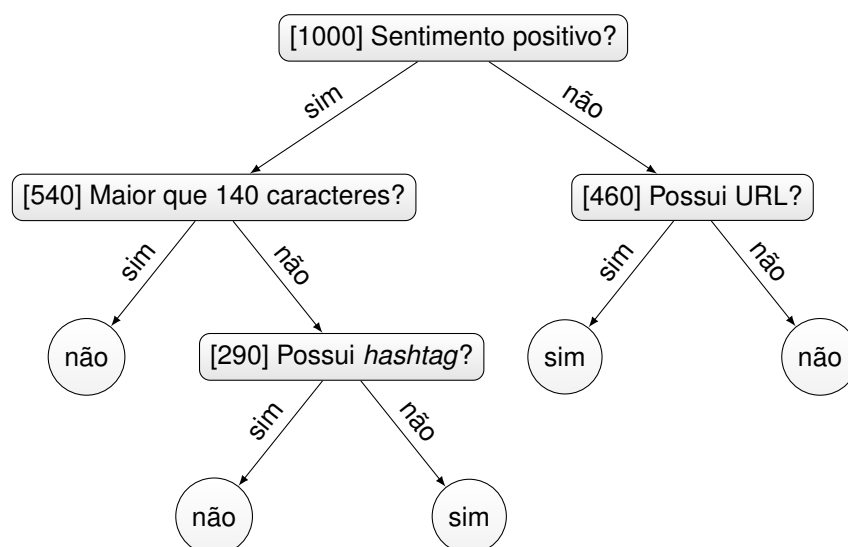
Abstraindo o conceito computacional, uma árvore de decisão pode ser definida por um fluxograma, no qual cada nó, com exceção do último nível, representa um teste

sobre as informações disponíveis. O ponto de partida é denominado nó raiz. A medida que os nós filhos vão sendo explorados, as informações são divididas com o objetivo de agrupá-las por similaridade e buscando o balanceamento entre os subgrupos. Ao percorrer toda a árvore, os últimos elementos, denominados nós folha, representam a decisão a ser tomada. Apesar de ser um conceito simples, a complexidade computacional desta técnica está no processo de indução da estrutura da árvore, feita de maneira automática e não-paramétrica, podendo lidar com dados multidimensionais.

Assim como outras técnicas dentro do escopo de aprendizagem supervisionada, árvores de decisão também são muito populares para a resolução de problemas de classificação de dados e regressão linear. Segundo (HAN; PEI; KAMBER, 2011), a popularização deste tipo de algoritmo na aprendizagem de máquina está diretamente ligada à sua característica não-paramétrica, que permite a indução de árvores sem a o total domínio ou configuração prévia dos dados, o que torna-se muito interessante no âmbito deste trabalho no que se refere à descoberta de maneira exploratória.

Ainda conforme (HAN; PEI; KAMBER, 2011), para realizar a classificação de dados, cada registro percorre um determinado caminho dentro da estrutura da árvore, partindo do nó raiz até o nó folha, o qual determina a classe para aquela entrada de dados. Para exemplificar, foi criada a árvore de decisão fictícia apresentada na Figura 2.4, a qual considera 4 atributos de uma mensagem de texto para considerá-la como popular ou não. Na figura, os nós com o formato retangular representam os testes feitos com cada registro de entrada, sendo que antes de cada teste é indicado o número de instâncias que estão naquele nó, partindo da raiz com 1000, já os nós com o formato circular são os nós folha, que representam a classificação final para indicar se a mensagem seria considerada como popular ou não.

Figura 2.4 – Exemplo de árvore para classificação de um tuíte como popular.



Fonte: Produção do próprio Autor.

Utilizando novamente como exemplo a frase “*With great power comes great responsibility*”, ao aplicá-la na árvore de decisão apresentada, ela seria classificada como popular, pois percorreria o seguinte caminho: Sentimento positivo; Menor que 140 caracteres; e não possui *hashtag*. Uma diferença em relação a técnica *Naive Bayes* que já é possível notar através deste exemplo, é que árvores de decisão são capazes de lidar com a correlação entre os atributos.

Mesmo existindo vários algoritmos com diferentes propostas para realizar a indução de árvores de decisão, duas etapas estão presentes na grande maioria deles durante o construção da árvore, a seleção das medidas de atributos e a “poda da árvore” que, respectivamente, são responsáveis por definir quais as melhores partições dos dados; e por remover, ou reduzir, ruídos nas ramificações gerados durante o treinamento. Apesar de serem etapas comuns na implementação deste tipo de algoritmo, também são as etapas que os diferenciam uns dos outros dependendo da abordagem adotada para realiza-las.

Um dos algoritmos que merece destaque por ser referência neste âmbito, é o CART (*Classification and Regression Trees*), criado em 1984 por um grupo de estatísticos (L. Breiman, J. Friedman, R. Olshen, and C. Stone). Ele realiza uma abordagem de construção recursiva de divisão e conquista partindo de cima para baixo, segundo (HAN; PEI; KAMBER, 2011).



### 3 PROPOSTA

A proposta deste trabalho é a elaboração de modelos, utilizando algoritmos de aprendizado de máquina supervisionado, capaz de classificar o nível de popularidade de tuítes com base na correlação entre a taxa de engajamento dos mesmos em função de um conjunto de características presentes no corpo das mensagens e o próprio texto. Para atingir esse objetivo, é necessário coletar os tuítes, extrair suas características e aplicar os algoritmos já mencionados para realizar o treinamento e classificação dos dados. A partir disso, esta seção apresenta a definição dos atributos, definição de popularidade, arquitetura de processamento dos tuítes e os classificadores utilizados na realização do trabalho.

#### 3.1 DEFINIÇÃO DOS ATRIBUTOS DE INTERESSE

Como parte do objetivo deste trabalho é a correlação entre a popularidade e as características do texto de cada tuíte, é de fundamental importância a definição e extração de características relevantes que possam influenciar no interesse dos usuários sobre uma determinada mensagem. Esta etapa corresponde a definição dos atributos que serão extraídos de cada um dos tuítes coletados. Os itens abaixo definem cada um destes atributos e a razão de terem sido escolhidos:

**Presença de URLs:** O uso de URLs em um tuíte pode indicar uma informação proveniente de outros meios, podendo ser sites de notícias ou outras mídias sociais, o que pode despertar, ou não, o interesse de usuários por um determinado tipo de informação. Esse atributo é representado pelo tipo de dados booleano, podendo ser verdadeiro ou falso.

**Presença de *hashtags*:** De maneira geral, as *hashtags* são palavras-chave ou termos utilizados para indicar que uma determinada mensagem está diretamente ligada a um tópico ou discussão específica. O que, de maneira semelhante ao uso de URLs, pode atrair o interesse de usuários por determinados tópicos. Este atributo também é do tipo booleano.

**Tamanho da mensagem:** Essa característica é basicamente a contagem da quantidade de caracteres usados no corpo do tuíte, que pode fazer com que os usuários percam o interesse em ler seu conteúdo, por ser muito curto ou muito extenso. Por tratar-se de um valor contínuo, este atributo é representado por um valor inteiro.

**Sentimento da mensagem:** O sentimento é um valor que classifica o teor do texto como positivo ou negativo. Fator que pode estar diretamente ligado a intenção

de cada usuário em propagar mensagens com um determinado humor. Este atributo também pode ser chamado de polaridade da mensagem e trata-se de um valor decimal, que pode variar entre -1 e 1, onde -1 corresponde a uma mensagem totalmente negativa, 0 corresponde a neutra e 1 corresponde a totalmente positiva.

**Banalidade da mensagem:** No contexto deste trabalho, como também em (OLIVEIRA; MERGEN, 2018), a banalidade corresponde à importância do que foi escrito no corpo do tuíte, levando em consideração a presença de palavras que são frequentemente usadas em textos escritos na língua inglesa. Sendo assim, quanto maior o número de palavras frequentes, mais banal é a mensagem. Este atributo é representado por um valor decimal, que varia entre 0 e 1, sendo que quanto mais próximo de 1, mais banal é a mensagem. O cálculo desta métrica utiliza a Equação 3.1, apresentada logo abaixo.

$$\frac{\sum_{i=1}^n (freq(P_i))}{n} \quad (3.1)$$

onde o conjunto  $\{P_1, \dots, P_n\}$  são as palavras da mensagem após a remoção de *stopwords* (preposições e artigos que normalmente são descartados durante o processamento de um texto). Já a função  $freq(P)$  retorna 1 caso a palavra  $P$  seja frequente e zero caso não seja.

### 3.2 DEFINIÇÃO DE POPULARIDADE

De maneira geral, em mídias sociais, a popularidade de uma conta pode ser medida através da quantidade de seguidores que ela detém, quanto maior o número de seguidores, mais influente, ou popular, a conta é considerada. Porém, este é um indicador simples que não determina o alcance real das publicações. Para isso, existem várias métricas que permitem uma medição mais precisa sobre o impacto causado pelas ações realizadas por uma determinada página ou usuário. Uma métrica muito conhecida e utilizada para medir o alcance real de uma página sobre seus seguidores é a taxa de engajamento. Esse índice considera as interações dos fãs com os conteúdos publicados, de forma que quanto maior é essa interação, maior é o nível de engajamento.

Como exposto em (PILLAT; PILLAT, 2017), para calcular a taxa de engajamento de uma determinada publicação, por convenção, é realizada a fórmula apresentada na Equação 3.2. Cada elemento da equação refere-se estritamente ao valor, em quantidade, obtido por cada publicação. Trazendo para a realidade do Twitter, os compartilhamentos são substituídos pelos retuítes e os comentários pelas respostas a um determinado tuíte.

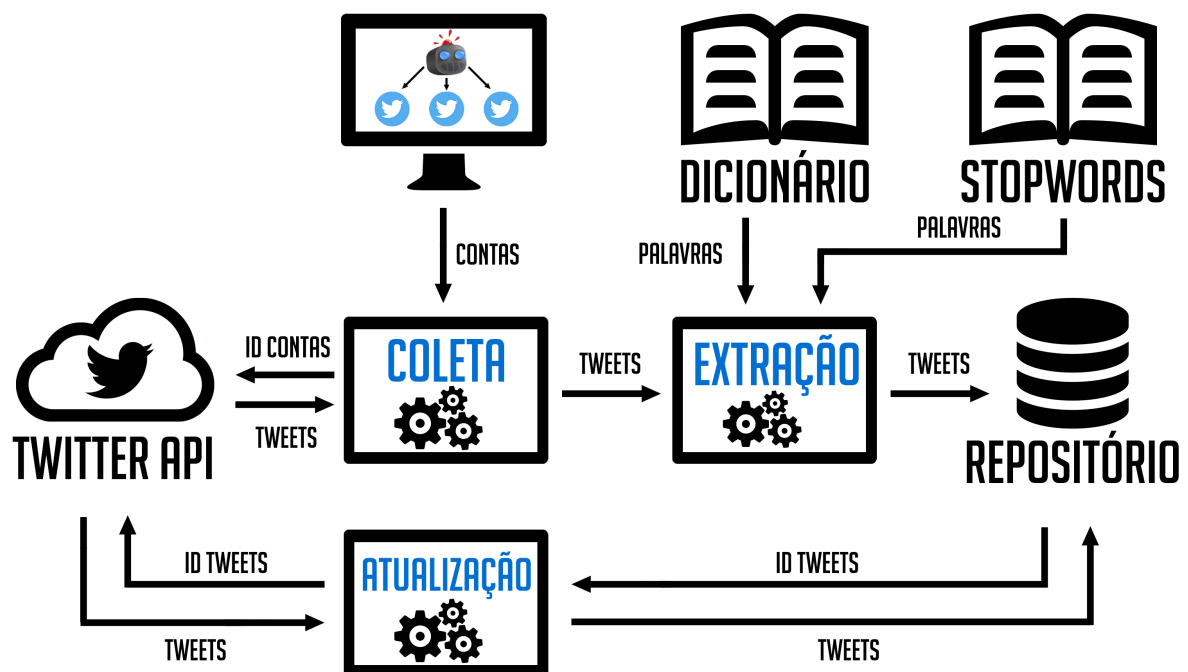
$$E(x) = \frac{curtidas + compartilhamentos + comentários}{seguidores} * 100 \quad (3.2)$$

Apesar de existir esta convenção para o cálculo do engajamento, a fórmula pode variar, dependendo das informações fornecidas por cada rede social. Como por exemplo, no caso do Facebook, o total de seguidores pode ser substituído pelo total de visualizações obtidas por cada publicação, ou então, como também apresentado em (PILLAT; PILLAT, 2017), substituído pelo seguidores da própria página mais os seguidores dos próprios fãs.

### 3.3 PROCESSAMENTO DOS TUÍTES

Esta seção engloba a descrição e detalhamento dos processos envolvidos na arquitetura adotada, apresentada na Figura 3.1, para realizar o processamento dos tuítes. Esta arquitetura, contém os seguintes módulos principais: (a) **Coleta** dos tuítes publicados por cada uma das contas acompanhadas; (b) **Extração** das características de cada tuíte; e (c) **Atualização** periódica dos dados coletados.

Figura 3.1 – Arquitetura adotada para extração de tuítes



Fonte: Produção do próprio autor.

A coleta dos tuítes é realizada tendo como base contas de personalidades influentes que utilizam o Twitter periodicamente. Ao todo, foram consideradas 30 contas

de diversas áreas de atuação, como por exemplo Donald J. Trump (atual presidente dos Estados Unidos), Jimmy Fallon (famoso apresentador de TV americano) e Katy Perry (cantora detentora da conta com o maior número de seguidores no Twitter).

A escolha deve-se ao fato de que a análise do impacto de publicações em redes sociais é mais relevante para esse tipo de usuário, uma vez o cálculo da taxa de engajamento para contas com poucos seguidores resultaria sempre em um valor próximo a zero. O que também é reforçado por (SUH et al., 2010), quanto maior a audiência, maiores são as chances de um tuíte ser retuitado. A coleta dos tuítes foi realizada durante o ano de 2018, totalizando cerca de 9500 registros distribuídos entre as contas de interesse que foram escritos na língua inglesa.

### 3.3.1 Coleta dos Tuítes

O módulo de coleta é responsável por extrair tuítes de usuários específicos. A extração ocorre de forma contínua, usando recursos de *streaming* disponibilizados pela API do Twitter (Twitter, 2018). São coletados todos tuítes publicados a partir do momento que o *streaming* entra em execução.

A especificação das contas a serem seguidas é feita através de uma conta raiz, a partir da qual são extraídos os tuítes publicados por todos usuários seguidos por esta conta. Esta estratégia permite que novas contas sejam adicionadas à lista sem que haja interrupções na execução do algoritmo. O módulo também conta com tratamento de exceções para que a coleta não seja interrompida devido à problemas temporários de acesso aos dados, como indisponibilidade do serviço ou extrapolação do limite de requisições permitido por instante de tempo.

Na Tabela 3.1 podem ser visualizadas as informações extraídas de cada tuíte através da API. O campo “mensagem” é usado para a extração das características. Já os campos “seguidores”, “retuítos” e “curtidas” são utilizados no cálculo para medir a taxa de engajamento de cada tuíte. Por sua vez, os campos “identificação” e “data/hora” são usados pelo módulo de atualização.

Tabela 3.1 – Dados coletados para cada tuíte

Informação	Conteúdo
autor	código e nome da conta que originou o tuíte
seguidores	quantidade de seguidores da conta que originou o tuíte
identificação	código do tuíte (permite a consulta posterior)
mensagem	texto de no máximo 280 caracteres
data e hora	data e hora da publicação do tuíte em seu país de origem
retuítos	quantidade de retuíte que a mensagem recebeu
curtidas	quantidade de vezes que o tuíte foi favoritado

Infelizmente a API do Twitter não permite a extração da quantidade de respostas na versão gratuita, apenas na versão para assinantes, impossibilitando a contabilização desse valor na fórmula de engajamento. Desta forma, a Equação 3.2, para o cálculo da taxa de engajamento, apresentada na seção 3.2, foi adaptada para considerar apenas as informações disponíveis, resultando na Equação 3.3.

$$E(x) = \frac{curtidas + retuítes}{seguidores} * 100 \quad (3.3)$$

### 3.3.2 Extração dos Atributos

Esta etapa corresponde a extração das características de cada um dos tuítes coletados. A extração ocorre imediatamente após a coleta. Os itens abaixo mostram como cada característica foi extraída:

**Presença de URLs e *hashtags*:** O uso desses recursos na mensagem é facilmente detectado pela presença de prefixos específicos no corpo da mensagem. Por exemplo, o prefixo “http” indica que URLs foram usadas, enquanto que o prefixo “#” denota o uso de *hashtags*.

**Tamanho da mensagem:** O tamanho é extraído através da contagem da quantidade de caracteres presentes no texto. A contagem desconsidera caracteres usados em URLs, assumindo que *hiperlinks* não transmitam nenhuma mensagem. A remoção de URLs foi realizada a partir da aplicação de uma expressão regular.

**Extração do sentimento:** Para realizar a extração do sentimento, foi utilizada a biblioteca TextBlob da linguagem Python (LORIA et al., 2014). Essa biblioteca permite a obtenção da polaridade e subjetividade de conteúdos textuais na língua inglesa. A API também fornece a possibilidade de tradução do conteúdo de textos escritos em outras linguagens. A extração do sentimento realizada pela biblioteca se baseia em Árvore de Decisão e no modelo de classificação *Naive Bayes* – ambos já apresentados na seção 2 –, o que elimina a necessidade de elaborar no novo algoritmo para realizar essa função.

**Extração da banalidade:** A verificação da frequência utiliza um dicionário contendo 3000 palavras comuns da língua inglesa<sup>1</sup>. Também são removidas as *hashtags* e menções a outros usuários, por entender que não se tratam de palavras que podem ser caracterizadas como banais ou não.

Na Tabela 3.2 pode ser visto um exemplo geral de todas as características extraídas nesta etapa.

<sup>1</sup>3000 most common words in English: <https://www.ef.com/english-resources/english-vocabulary/top-3000-words/>

Tabela 3.2 – Dados obtidos na etapa de Extração

Informação	Conteúdo
sentimento	valor entre -1 e 1 correspondente a polaridade do texto
URL	valor 1 se houver URL no texto e 0 se não houver
<i>hashtag</i>	valor 1 se houver <i>hashtag</i> no texto e 0 se não houver
tamanho	quantidade de caracteres utilizados na mensagem
banalidade	somatório baseado na no uso de palavras frequentes

### 3.3.3 Atualização dos Dados de Retuítas e Curtidas

Como o módulo de coleta funciona por meio de *streaming*, os tuítes são coletados no instante de sua criação. Nesse momento, a quantidade de retuítas e curtidas recebidos têm o valor zero. Dessa forma, é necessária uma conferência periódica para a obtenção dos dados atualizados.

A atualização é realizada através de um recurso da API do Twitter que obtém informações de um tuíte a partir do seu código de identificação. Para evitar sobrecarga de processamento, apenas os tuítes publicados no intervalo de 15 dias são atualizados. Como os dados de tuítes mais antigos raramente são modificados, a busca para a atualização de cada um deles seria ao mesmo tempo custosa e improdutiva.

## 3.4 CLASSIFICAÇÃO DOS TUÍTES

Esta etapa descreve os processos e ferramentas utilizados na realização da classificação dos tuítes como populares ou não populares. Para a realização da predição dos tuíte tendo como base a taxa de engajamento e considerando os atributos já mencionados, serão utilizados os algoritmos de classificação já apresentados Naive Bayes e Árvores de decisão. Os experimentos com cada algoritmo são realizados através da implementação de código utilizando a linguagem de programação Python (Python Software Foundation, 2018) ou com o auxílio da ferramenta Weka (The University of Waikato, 2018).

A utilização da linguagem Python é justificada pela grande quantidade de bibliotecas que proporcionam maior facilidade em lidar com a manipulação de dados e aprendizado de máquina. Além disso, a linguagem conquistou grande popularidade dentre a comunidade que trabalha com inteligência artificial. Dentre estas bibliotecas, uma delas merece ser destacada aqui, que é a SciKit-Learn (Scikit-Learn Community, 2018), que consiste em um conjunto de funcionalidades específicas para trabalhar com diferentes modelos de aprendizado de máquina, dentre elas classificação, regressão, agrupamento, redução de dimensionalidade e pré-processamento.s

O Weka (sigla em inglês para *Waikato Environment for Knowledge Analysis* e também nome de uma ave da Nova Zelândia) é uma ferramenta desenvolvida na linguagem Java que oferece um ambiente preparado para auxiliar no processo de análise e mineração de dados. Esse ambiente provê uma coleção de algoritmos capazes para realizar tarefas como preparação de dados, classificação, regressão, agrupamento, mineração de regras de associação e visualização (The University of Waikato, 2018).

### 3.4.1 Balanceamento das instâncias

Como apresentado em (HAN; PEI; KAMBER, 2011), a maioria dos modelos tradicionais de aprendizagem de máquina consideram que os dados de entrada já estão bem distribuídos dentre as classes, o que geralmente não acontece em bases de dados reais. Para resolver este problema, existem diversas técnicas para aperfeiçoar a classificação com dados desbalanceados, duas destas dela são: *oversampling*, que consiste no preenchimento dos dados da classe com menor número até que ambas sejam equivalentes; e o *undersampling*, que consiste na redução dos dados da classe com o maior número.

Desta forma, para realizar o treinamento e a validação dos dados, independente do modelo de aprendizagem aplicado, é importante que as entradas estejam bem distribuídas entre as classes. O desbalanceamento pode resultar em um modelo tendencioso, havendo a probabilidade de classificar os dados como uma determinada classe devido à predominância da mesma no momento do treinamento.

Além do balanceamento, é preciso preparar a base de dados para cada teste, isto é, configurar os dados de entrada para cada modelo considerando a taxa de engajamento e o usuário autor dos tuítes. Para realizar este processo, foi elaborado uma algoritmo que divide os registros igualmente entre as duas classes considerando uma determinada taxa de engajamento e autor passados por parâmetro. Este processo também foi escrito na linguagem Python e uma versão simplificada, em pseudo código, pode ser vista logo abaixo no algoritmo 1.

O algoritmo tem como entrada a taxa de engajamento e o código do autor, que pode não ser informado, para o caso de análises gerais. Como primeira instrução, é feita uma busca pelos tuítes considerando a taxa e o usuário. A própria função de busca faz o tratamento da necessidade de distinção dos dados por usuário. Tendo a lista completa, é identificada a classe que possui o menor número de registros, sendo esta a variável responsável pelo balanceamento. São inicializadas as variáveis de controle (*nPop* e *nNaoPop*), *dados* e *classes*. Em seguida é feito um laço de repetição para cada tuíte. Para cada iteração do laço é feita uma verificação da quantidade de entradas em cada classe através das variáveis de controle e, se o valor for menor

---

**Algoritmo 1** Algoritmo para preparação dos dados
 

---

```

1: function PREPARADADOS(taxa, autor = 0)
2:   twites  $\leftarrow$  embaralha(buscaListaDeTwites(taxa, autor))
3:
4:   max  $\leftarrow$  maximoPorClasse(twites)
5:   nPop, nNaoPop  $\leftarrow$  0, 0
6:   dados, classes  $\leftarrow$  [], []
7:
8:   for each tw in twites do
9:     if (nPop == max E tw['pop']) OU (nNaoPop == max E !tw['pop']) then
10:      popular
11:
12:      dados.novo(tw['atributos'])
13:      classes.novo(tw['pop'])
14:
15:      if tw['popular'] then
16:        nPop  $\leftarrow$  nPop + 1
17:      else
18:        nNaoPop  $\leftarrow$  nNaoPop + 1
19:
20:      if quantidade(dados) == max * 2 then
21:        parar
22:  return dados, classes

```

---

que o máximo, os atributos e classe são guardados. Por fim, as variáveis de controle são incrementadas. O laço termina quando a quantidade de dados alcança o valor máximo, retornando então as listas balanceados.

O código descrito no Algoritmo 1 é uma versão reduzida. Há ainda o trecho responsável pela geração do arquivo para utilização no Weka, se necessário for. O formato deste arquivo gerado pelo código e aceito pela ferramenta é o ARFF (sigla em inglês para *Attribute-Relation File Format*).

### 3.4.2 Classificação utilizando Naive Bayes e Árvores de Decisão

Como já mencionado, a classificação dos dados será realizada utilizando os algoritmos Naive Bayes e Árvores de Decisão. No caso do modelo de Bayes para classificação binária, os experimentos são realizados em duas vertentes, sendo uma delas a aplicação do algoritmo utilizando somente o texto pré-processado dos tuítes e a outra utilizando os atributos coletados.

O modelo Naive Bayes aplicado ao texto foi implementado utilizando a linguagem Python, com o auxílio da biblioteca para aprendizado de máquina Scikit-Learn.



O algoritmo é baseado em cálculos probabilísticos conforme apresentado na seção 2.3.1 e, assim como código de balanceamento exposto, este também recebe como parâmetros de entrada a taxa de engajamento e o usuário autor. Ao final da execução, como retorno são apresentadas as métricas de avaliação do modelo, juntamente com a matriz de confusão. Esse formato de saída foi adotado com o intuito de manter a consistência com os resultados obtidos a partir dos testes utilizando o Weka.

No caso dos experimentos utilizando os atributos coletados, tanto a segunda opção do Naive Bayes quanto as Árvores de Decisão, estes são realizados utilizando o Weka, devido sua praticidade em realizar o treinamento e validação dos dados a partir do arquivo gerado. Quanto aos modelos de árvores, os testes realizados focam nos algoritmos J48 e LTM (sigla para *Logistic Model Trees*).

Para a geração das bases utilizadas nos experimentos com o Weka, a saída do algoritmo de balanceamento é gravada em um arquivo do tipo ARFF, no qual também são descritos, além dos próprios dados, os atributos e seus tipos. Cada arquivo, que tem como variante a taxa de engajamento e o usuário, é carregado na ferramenta e aplicado para cada um dos três algoritmos. Com os resultados de cada execução, são gerados gráficos para uma análise visual do desempenho de cada modelo. As análises mais significativas são mostradas na seção 4 a seguir.

## 4 EXPERIMENTOS

Com base nos fundamentos e proposta apresentados, respectivamente nas seções 2 e 3, neste capítulo serão apresentados os experimentos realizados e resultados obtidos com a aplicação dos algoritmos sugeridos sobre os dados coletados. De maneira geral, os experimentos tem por objetivo realizar análises sobre a aplicação dos algoritmos de aprendizado de máquina, variando o usuário e a taxa de engajamento para determinar a popularidade. O intuito é identificar uma possível correlação entre estas variantes e as características extraídas de cada tuíte.

### 4.1 MÉTRICAS DE AVALIAÇÃO

A fim de realizar a avaliação dos modelos de avaliação, é importante que sejam definidas as métricas utilizadas na comparação entre os modelos. Para isso, serão consideradas cinco métricas, sendo acurácia, sensibilidade, especificidade, valor preditivo positivo e valor preditivo negativo. Estas medidas, apresentadas respectivamente nas equações 4.1, 4.2, 4.3 e 4.4, retiradas e traduzidas de (HAN; PEI; KAMBER, 2011), são aplicadas após a etapa de validação. Elas consideram os acertos e erros do algoritmo sobre cada uma das classes em relação ao total de cada classe.

A **acuracidade** é uma medida de análise geral dos acertos, sem diferenciação entre as classes. Enquanto que a **sensibilidade** e a **especificidade** são as medidas que indicam a capacidade do modelo em realizar a predição das entradas como a classe positiva e negativa, respectivamente. Já os valores **preditivos**, expressa a relação do total de predições corretas de uma das classes com o total de predições realizadas para essa mesma classe. As formulas de valor preditivo negativo e positivo, assim como a sensibilidade e especificidade, são equivalentes, variando apenas a classe em questão.

$$Acurácia = \frac{AcertosPositivos + AcertosNegativos}{TotalPositivos + TotalNegativos} \quad (4.1)$$

$$Sensibilidade = \frac{AcertosPositivos}{TotalPositivos} \quad (4.2)$$

$$Especificidade = \frac{AcertosNegativos}{TotalNegativos} \quad (4.3)$$

$$PreditivoPositivo = \frac{AcertosPositivos}{AcertosPositivos + FalsosPositivos} \quad (4.4)$$

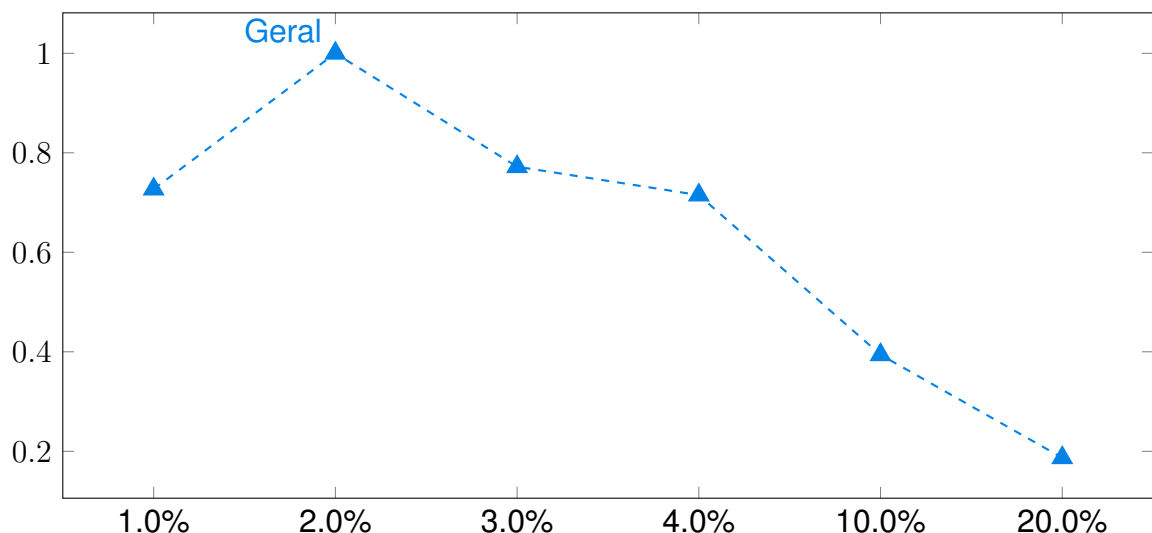
$$PreditivoNegativo = \frac{AcertosNegativos}{AcertosNegativos + FalsosNegativos} \quad (4.5)$$

Apesar da importância da utilização de todas as métricas apresentadas, destaca-se a importância de duas delas, a acurácia e a sensibilidade. A acurácia por expressar o acerto do modelo de maneira geral, e a sensibilidade por expressar o acerto do modelo sobre a classe de interesse, que é o caso de um tuíte ser positivo. Desta forma, pode-se estabelecer como prioridade a otimização dos resultados obtidos nestas medidas.

## 4.2 APLICAÇÃO DOS CLASSIFICADORES

No intuito de analisar a relação entre a quantidade de instâncias balanceadas para cada classe e a variação da taxa de engajamento, foi elaborado o gráfico 4.1. Por questões de otimizar a visualização das demarcações entre as contas, as quantidades foram normalizadas, assumindo como 100% os valores os valores que maximizam o balanceamento das classes. A partir do gráfico gerado, é possível observar que o valor ótimo para maximizar o número de dados contidos em cada classe é 2%, se consideradas todas as contas, e a variação dessa taxa para qualquer direção, resulta em uma perda de dados considerável.

Figura 4.1 – Variação no número de instâncias balanceadas por taxa de engajamento.

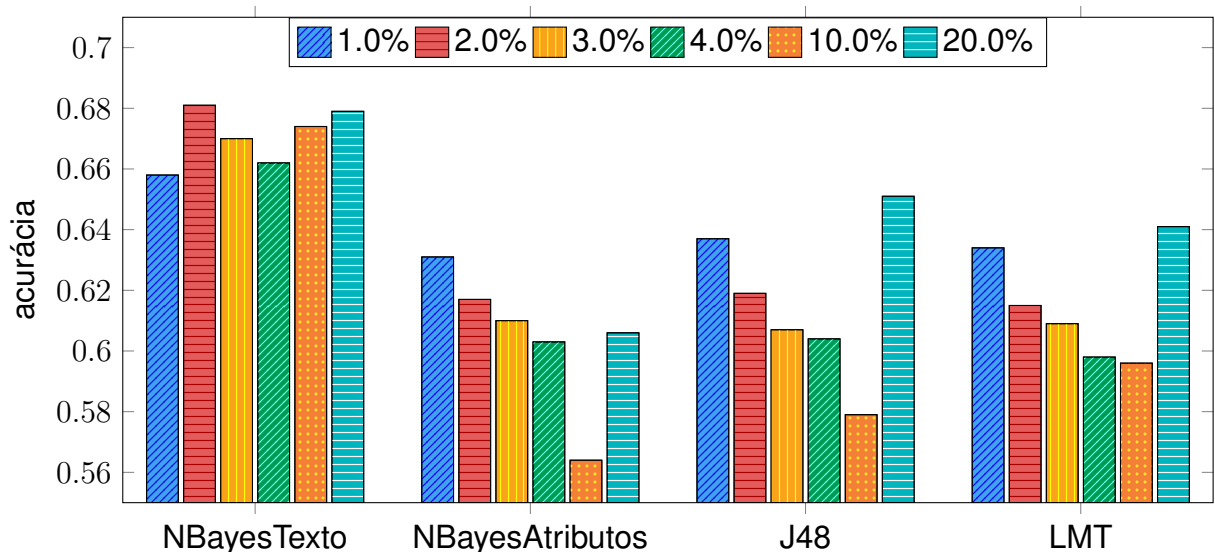


Fonte: Produção do próprio autor.

No gráfico 4.2 a seguir, é exibida a variação da acurácia de cada um dos quatro modelos testados conforme a taxa de engajamento é modificada. Neste experimento são considerados os tuítes de todos os usuários da base de dados. É possível iden-

tificar que o algoritmo Naive Bayes aplicado ao texto apresentou resultados melhores em relação aos demais, que foram aplicados utilizando os atributos extraídos de cada mensagem. O melhor resultado obtido neste modelo foi utilizando a taxa de engajamento de 2%, que alcançou 68.1% de acurácia. Apesar de apresentar um melhor resultado neste experimento, todos os algoritmos obtiveram valores muito próximos, sendo 56.4% o menor deles – caso do Naive Bayes aplicado aos atributos e utilizando a taxa de engajamento de 10%.

Figura 4.2 – Acurácia de cada algoritmo aplicado sobre toda a base com diferentes taxas de engajamento.

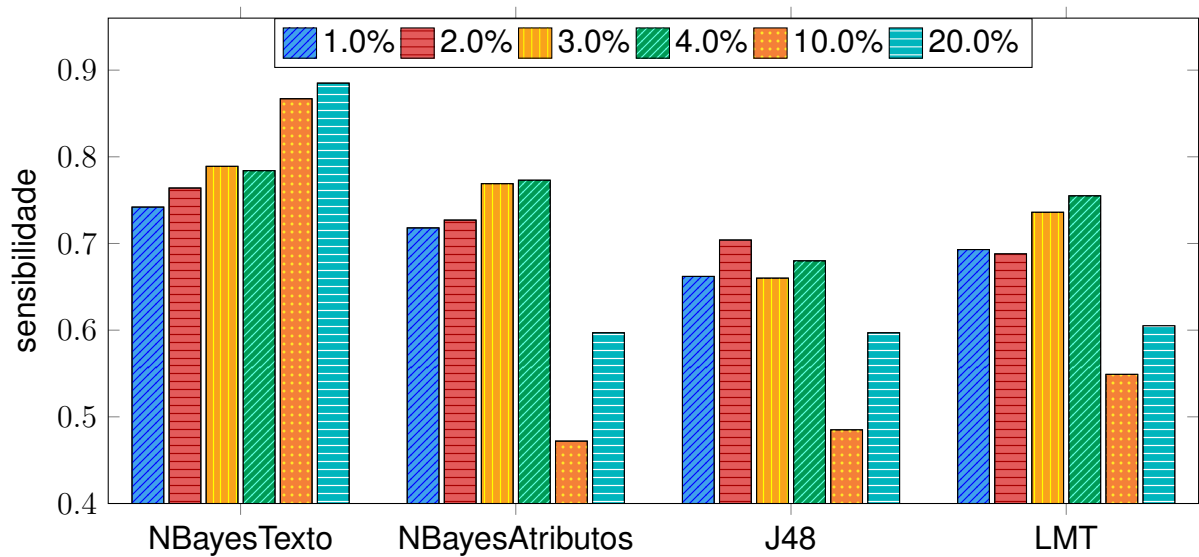


Fonte: Produção do próprio autor.

Ainda realizando a análise sobre os dados gerais de todas as contas e, dando atenção para outra métrica de interesse, foi elaborado o gráfico 4.3, o qual é equivalente ao 4.2 apresentado logo acima. Neste caso, é exposta a variação no valor desta métrica, sensibilidade, para cada algoritmo e considerando a modificação na taxa de engajamento. Novamente, apesar de intervalos muito próximos, nota-se que o modelo Naive Bayes aplicado ao texto obteve resultados melhores que os demais que foram aplicados sobre os atributos. No melhor dos resultados, este modelo obteve 88.5% no valor de sensibilidade e, curiosamente, utilizando como taxa de engajamento o valor de 20%.

Com o objetivo de tentar identificar uma possível relação entre a variação no número de instâncias e as métricas de acurácia, sensibilidade e especificidade, foi elaborado o gráfico 4.4, exibido logo abaixo, no qual foram delineadas as curvas de cada medida para o modelo Naive Bayes aplicado ao texto, além da própria curva do balanceamento das instâncias – também normalizado, assim como no gráfico 4.1. Observa-se inicialmente, que não há uma relação nítida entre as curvas. Porém,

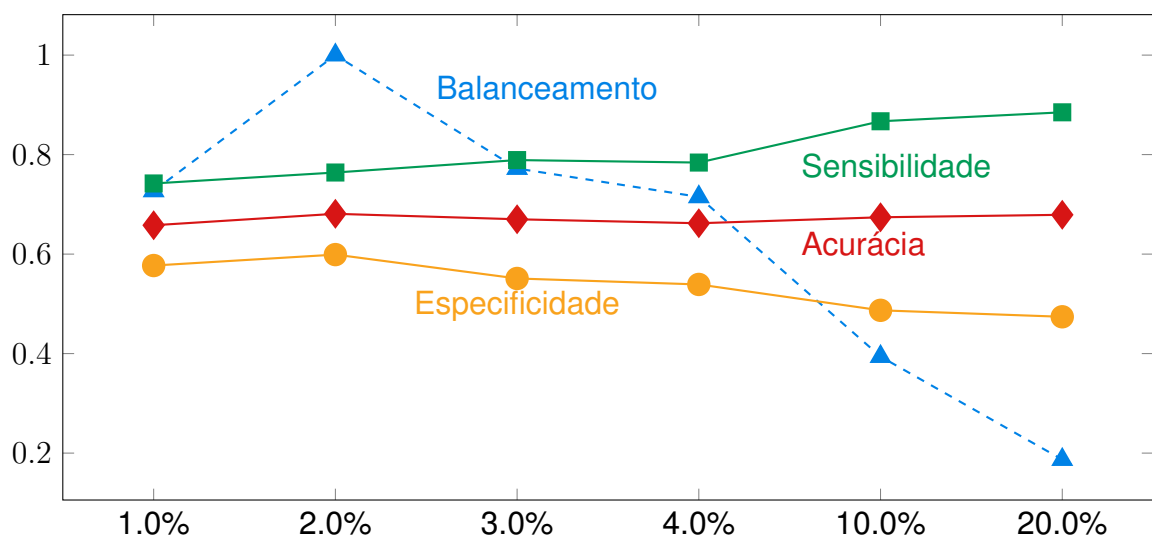
Figura 4.3 – Sensibilidade de cada algoritmo aplicado sobre toda a base com diferentes taxas de engajamento.



Fonte: Produção do próprio autor.

isso pode ser atribuído ao fato de que a variação no intervalo das métricas é muito pequeno. Apesar disso, considerando também o gráfico 4.2, observa-se que a melhor acurácia, também é aquela que maximiza o balanceamento entre as classes. O mesmo padrão não aplica-se ao caso da sensibilidade, que aparentemente tem a tendência de aumentar, conforme o número de instâncias é reduzido. Em contrapartida, a curva de especificidade aparentemente tem a melhor relação com o balanceamento das instâncias, já que tem como melhor resultado a taxa de 2% e tende a diminuir ao variar essa taxa para qualquer direção.

Figura 4.4 – Variação no número de instâncias em relação às métricas para o algoritmo Naive Bayes utilizando o texto.



Fonte: Produção do próprio autor.

## **5 CONCLUSÕES**

Considerações finais e trabalhos futuros.

## REFERÊNCIAS BIBLIOGRÁFICAS

BENEVENUTO, F. et al. Detecting spammers on twitter. In: **Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)**. [S.l.: s.n.], 2010. v. 6, n. 2010, p. 12.

DUAN, Y. et al. An empirical study on learning to rank of tweets. In: **Proceedings of the 23rd International Conference on Computational Linguistics**. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010. (COLING '10), p. 295–303. Disponível em: <<http://dl.acm.org/citation.cfm?id=1873781.1873815>>.

HAN, J.; PEI, J.; KAMBER, M. **Data Mining: Concepts and Techniques**. Elsevier Science, 2011. (The Morgan Kaufmann Series in Data Management Systems). ISBN 9780123814807. Disponível em: <<https://books.google.com.br/books?id=pQws07tdpjoC>>.

KHARDE, V. A.; SONAWANE, S. Sentiment analysis of twitter data : A survey of techniques. **CoRR**, abs/1601.06971, 2016. Disponível em: <<http://arxiv.org/abs/1601.06971>>.

LORIA, S. et al. Textblob: simplified text processing. **Secondary TextBlob: Simplified Text Processing**, 2014.

NAVEED, N. et al. Bad news travel fast: A content-based analysis of interestingness on twitter. In: **Proceedings of the 3rd International Web Science Conference**. New York, NY, USA: ACM, 2011. (WebSci '11), p. 8:1–8:7. ISBN 978-1-4503-0855-7. Disponível em: <<http://doi.acm.org/10.1145/2527031.2527052>>.

OLIVEIRA, L. L. de; MERGEN, S. L. S. Análise da popularidade de tuítes com base em características extraídas de seu conteúdo. **Escola Regional de Banco de Dados (ERBD)**, v. 14, n. 1/2018, 2018. ISSN 2595-413X. Disponível em: <<http://portaldeconteudo.sbc.org.br/index.php/erbd/article/view/2834>>.

PILLAT, V. G.; PILLAT, V. G. Comparação entre duas fórmulas utilizadas para o cálculo da taxa de engajamento utilizando como base a porcentagem de visualizações e o total de fãs. **Revista Brasileira de Pesquisas de Marketing**, 2017. ISSN 2317-0123. Disponível em: <[http://www.revistapmkt.com.br/pt-br/anteriores/anteriores.aspx?udt\\\_863\\\_param\\\_detail=8650](http://www.revistapmkt.com.br/pt-br/anteriores/anteriores.aspx?udt\_863\_param\_detail=8650)>.

Python Software Foundation. **Python 3.7.1 documentation**. Python Software Foundation, 2018. Acessado em nov 2018. Disponível em: <<https://docs.python.org/3/>>.

RUSSELL, S.; NORVIG, P. **Inteligência artificial: Tradução da 3a Edição**. Elsevier Editora Ltda., 2014. 1056 p. ISBN 9788535251418. Disponível em: <<https://books.google.com.br/books?id=BsNeAwAAQBAJ>>.

Scikit-Learn Community. **Scikit-Learn: machine learning in Python**. Scikit-Learn Community, 2018. Acessado em nov 2018. Disponível em: <<https://scikit-learn.org/stable/>>.

SUH, B. et al. Want to be retweeted? large scale analytics on factors impacting retweet in twitter network. In: IEEE. **2010 IEEE Second International Conference on Social**

**Computing**. 2010. p. 177–184. Disponível em: <<http://ieeexplore.ieee.org/document/5590452/>>.

The University of Waikato. **Weka 3 - Data Mining with Open Source Machine Learning Software in Java**. The University of Waikato, 2018. Acessado em nov 2018. Disponível em: <<https://www.cs.waikato.ac.nz/ml/weka/>>.

Twitter. **Twitter Developer Platform - Twitter Developers**. Twitter, 2018. Acessado em nov 2018. Disponível em: <<https://developer.twitter.com/>>.

XU, Z.; YANG, Q. Analyzing user retweet behavior on twitter. In: **2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining**. [S.l.: s.n.], 2012. p. 46–50.

ZHENG, A.; CASARI, A. **Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists**. [S.l.]: "O'Reilly Media, Inc.", 2018.