UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

GABRIEL HENRIQUE DANIEL DA SILVA

Extração, Classificação e Priorização de Reclamações de Consumidores em SACs Online Baseados em Texto

Recife

2023

GABRIEL HENRIQUE DANIEL DA SILVA

Extração, Classificação e Priorização de Reclamações de Consumidores em SACs Online Baseados em Texto

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Informática (CIn) da Universidade Federal de Pernambuco como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Orientadora: Drª Flávia de Almeida Barros

Recife

2023

Catalogação na fonte
Bibliotecária Nataly Soares Leite Moro, CRB4-1722

S586e Silva, Gabriel Henrique Daniel da
Extração, classificação e priorização de reclamações de consumidores em SACs online baseados em texto / Gabriel Henrique Daniel da Silva – 2023.
78 f.: il., fig., tab.

Orientadora: Flávia de Almeida Barros.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.
Inclui referências.

1. Inteligência computacional. 2. SAC. 3. Extração de informação. 4. Aprendizagem de máquina. 5. Classificação. I. Barros, Flávia de Almeida (orientadora). II. Título

006.31 CDD (23. ed.) UFPE - CCEN 2023 – 191

Gabriel Henrique Daniel da Silva

“Extração, Classificação e Priorização de Reclamações de Consumidores em SACs Online Baseados em Texto”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 28 de agosto de 2023.

BANCA EXAMINADORA

_____

Profa. Dra. Patrícia Cabral de Azevedo Restelli Tedesco
Centro de Informática / UFPE

_____

Prof. Dr. Péricles Barbosa Cunha de Miranda
Departamento de Computação / UFRPE

_____

Profa. Dra. Flávia de Almeida Barros
Centro de Informática / UFPE
(Orientadora)

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado o dom da vida, por ter me dado a graça de nascer numa família tão maravilhosa e por estar sempre conosco em todos os momentos.

Em segundo lugar, a minha família, as cinco pessoas que eu mais amo no mundo, por seu apoio, suporte e amor incondicional. Sem dúvida, eu não seria nada do que eu sou hoje, se não fosse por causa de vocês.

À minha orientadora, professora Flávia, por sua paciência e disponibilidade. Seu auxílio e orientação foi de vital importância para que fosse possível concluir este trabalho.

À todos os professores do CIn que de alguma forma contribuíram com minha formação, desde a graduação até a conclusão desta dissertação.

“Tenha fé, vá na fé, nunca perca a fé em Deus.” (NOGUEIRA, 2012)

RESUMO

Os serviços de atendimento ao consumidor (SACs) são canais de comunicação entre uma empresa e seus consumidores, possibilitando que os clientes tirem dúvidas, deem sugestões, solicitem informações e registrem reclamações. Com o advento da Internet e das redes sociais, grande parte do contato é feito online através de canais descentralizados que geram uma enorme quantidade de informação textual a ser analisada. Consumidores insatisfeitos e que não se sentem priorizados tendem a se afastar e tomar ações que possam influenciar negativamente na imagem da empresa. Nesse contexto, este trabalho de mestrado propõe um processo para auxiliar as empresas a lidar com essa grande quantidade de reclamações que ficam expostas na Web através do processamento automático dos textos das reclamações. A solução proposta se baseia em uma estratégia de extração automática das reclamações postadas pelos consumidores em sites específicos, seguida de classificação e ranqueamento dessas reclamações, a fim de priorizar as críticas consideradas mais relevantes (i.e., com maior potencial de prejuízo) no momento. O processo proposto foi implementado em três etapas distintas. Inicialmente, foi desenvolvido um módulo para criação do corpus que realiza um *scrapping* para extração das reclamações a partir de sites na Web. O código foi desenvolvido em Python com o auxílio do framework Selenium. O site escolhido para o protótipo inicial foi o “ReclameAQUI”. A partir dos dados extraídos, foi criado um corpus contendo reclamações que foram etiquetadas manualmente por pessoas com experiência no domínio de atendimento ao consumidor, também foi realizado um pré-processamento textual. A seguir, foi desenvolvido um classificador de texto baseado em Aprendizagem de Máquina usando o corpus etiquetado. Foram realizados diversos experimentos buscando encontrar a combinação de melhor desempenho dentre as opções disponíveis. A configuração que utiliza *TF-IDF* para transformação de texto, *K-Fold Cross Validation* no treinamento e *Regressão Logística* teve o melhor resultado, com acurácia de 82,22%, F-measure de 82,39% e área sob a curva ROC de 0,8881. Por fim, o protótipo implementado também realiza o ranqueamento das reclamações prioritárias, oferecendo ainda a possibilidade de exportação das reclamações de forma ordenada.

Palavras-Chave: SAC; extração de informação; aprendizagem de máquina; classificação.

ABSTRACT

The customer service management (SCMs) are communication channels between a company and their customers. It allows their customers to ask questions, make suggestions, request information and register complaints. With the advent of the internet and social networks, a major part of that contact is now made online by multiple channels which generate a huge quantity of text information to be handled. Dissatisfied customers may cut ties and take actions that can cause negative influence on a company's image. In this context, this paper proposes a process to help companies to handle the huge amount of complaints which are displayed on the web by automatically processing those complaints. The proposed solution is based on a strategy for automatic extraction of open customer's complaints at specific websites, followed by classification and ranking of those complaints to prioritize the ones considered most relevant (i.e., with most potential waste) at the time. The proposed process has three distincts modules. The first module was responsible for creating a corpus by making a scrapper to extract the complaints from websites. The code was made in Python with the Selenium Framework. The choosed website for the initial prototype is "ReclameAQUI". The extracted data was manually labeled by humans with experience with customer service. The second module was responsible for pre-processing the text. The last module develops a machine learning based text classifier using the labeled corpus. A few experiments were carried out to search for the best performance between the available options. The one usings *TF-IDF* for text transformation, *K-Fold Cross Validation* on training and *Logistic Regression* obtained the best result with 82.22% of accuracy, 82.39 of F-measure and 0.8881 of AUC. The implemented prototype ranks the complaints that were classified as significant. It has also the option to export the ordered complaints.

Keywords: SCM; information extraction; machine learning; classification.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AM	Aprendizagem de Máquina
AUC	Área Sob a Curva
BoW	Bag of Words
CSV	Comma-separated values
KDD	Knowledge Discovery in Databases
KDT	Knowledge Discovery from Text
KNN	K-Nearest Neighbors
MLP	MultiLayer Perceptron
MT	Mineração de Texto
PLN	Processamento de Linguagem Natural
RBF	Radial Basis Function
RNA	Redes Neurais Artificiais
ROC	Receiver Operating Characteristic
SAC	Serviço de Atendimento ao Consumidor
SVM	Support Vector Machine
TF-IDF	Term-Frequency – Inverse Document Frequency

SUMÁRIO

1 INTRODUÇÃO

Nos últimos anos, a internet vem se popularizando cada vez mais no Brasil trazendo consigo uma mudança de paradigma para diversos setores de grandes empresas que trabalhavam de forma diferente antes dessa mudança. Dentre os setores que sofreram mudanças com o advento da internet está o Serviço de Atendimento ao Consumidor (SAC).

Os SACs são canais de comunicação entre uma empresa e seus consumidores. Eles possibilitam aos consumidores tirar dúvidas, dar sugestões, solicitar informações sobre produtos ou serviços, registrar reclamações entre outras atividades relacionadas.

A presença de um bom serviço de atendimento ao consumidor está ligada a um aumento no nível de satisfação percebido nos consumidores. (SANTOURIDIS; VERAKI, 2017 apud FEINBERG & KADAM, 2002). Inclusive, podemos afirmar que entre os principais objetivos de um serviço de atendimento ao consumidor estão justamente a capacidade de alcançar a satisfação de seus clientes a longo prazo, bem como obter lucros organizacionais (ABDULLATEEF; SALLEH, 2013 apud COLTMAN, 2007; EID, 2007; KOHLI ET AL., 1993; SIN ET AL., 2005).

A satisfação do consumidor está diretamente relacionada à noção de lealdade dos consumidores. Nesse contexto, um serviço de atendimento ao consumidor eficaz acaba influenciando na capacidade de uma empresa de gerar consumidores leais à mesma (HARYANDIKA; SANTRA, 2021).

Garantir a lealdade de seus clientes é uma das principais ambições da maioria empresas, uma vez que, consumidores leais, tendem a comprar mais, gastar mais e ser menos rigoroso com os preços ofertados (DE LEANIZ; DEL BOSQUE RODRÍGUEZ, 2016). Chegando inclusive, ao ponto de encontrar correlação entre lucratividade e lealdade de seus clientes (BOWEN; CHEN, 2001).

Empresas capazes de melhorar a imagem com a qual são percebidas tendem a melhorar a sua reputação e facilitar o processo de construção de lealdade nos seus consumidores (DE LEANIZ; DEL BOSQUE RODRÍGUEZ, 2016).

A imagem de uma empresa pode ser definida como “O sentimento e as crenças existentes sobre uma determinada empresa que está presente na mente

das pessoas” (BERNSTEIN, 1992). Apesar da definição de imagem de uma empresa ser relacionada a um conceito abstrato e que pode variar bastante através de fatores externos de acordo com a percepção das pessoas, isso não significa que a mesma não possa ser gerenciada pela própria empresa. Seja aplicando técnicas de marketing ou através da postura aplicada em canais de comunicação, sejam eles internos ou externos à empresa (DA CAMARA, 2011). Além dos fatores citados anteriormente, quaisquer canais que possibilitem troca direta de experiências entre consumidores, opiniões e o próprio boca-a-boca são fatores capazes de influenciar na imagem que uma empresa passa para seus clientes (KELLER, 2003).

Electronic word-of-mouth, ou boca-a-boca eletrônico, acaba servindo como uma fonte de informação alternativa aos consumidores. Dessa forma, percebe-se quão poderosa essa ferramenta pode ser, com potencial para influenciar no processo decisório de consumo e inclusive limitar os efeitos causados por campanhas de marketing e publicidade (JALILVAND, 2011).

Sites como o Reclame Aqui e o Consumidor.gov são exemplos de portais que possibilitam um contato público e direto entre o consumidor e a empresa dona do produto ou serviço citado no contato para reclamação.

As reclamações registradas pelos consumidores ficam armazenadas num repositório público, podendo ser consultadas por qualquer pessoa. Dessa forma é vital para a boa imagem das empresas perante o público, que o máximo possível de reclamações sejam atendidas e resolvidas de forma satisfatória aos consumidores. Uma vez que empresas com imagens consideradas positivas possuem uma maior facilidade de conquistar a lealdade de seus clientes, além de em diversos casos se tornar um diferencial competitivo. (DE LEANIZ; DEL BOSQUE RODRÍGUEZ, 2016).

Além de prezar por uma boa imagem, as empresas também se preocupam em tentar manter na sua base de clientes, aqueles que estão insatisfeitos ou ao menos minimizar o nível de frustração causado aos que efetivamente registraram reclamações. Uma vez que segundo Hawkins, Mothersbaugh & Best (2007) quando lidamos com clientes que se sentem insatisfeitos e que decidem agir sobre esse sentimento, a consequência que poderia ser considerada como mais vantajosa ou analisando por outro ponto de vista, menos prejudicial para empresa, seria justamente a abertura de uma reclamação.

Para tentar evitar que a insatisfação escale para ações ainda mais danosas a empresa que o registro de uma reclamação, surge a necessidade de se aprofundar um pouco mais no assunto, para tanto, elaboramos as seguintes questões de pesquisa que nortearam o presente trabalho.

1.1 QUESTÃO DE PESQUISA E TRABALHO REALIZADO

Durante as pesquisas realizadas e o processo de execução deste trabalho, as seguintes questões de pesquisa foram sendo respondidas:

1. Quais estratégias seriam mais adequadas para minimizar a perda de clientes que efetuaram o registro de reclamações na web?
2. Como identificar automaticamente as reclamações mais relevantes considerando os dados disponíveis?
3. Qual algoritmo de aprendizagem de máquina é capaz de classificar as reclamações textuais como prioritárias com a melhor performance de acordo com as métricas de precisão, cobertura, acurácia, f-measure e curva roc?
4. Um modelo com menos dados, porém de um ramo de negócio específico traz resultados melhores que um contendo reclamações de empresas com naturezas diversas?

O objetivo principal deste trabalho foi definir um processo para construção de aplicações capazes de extrair, classificar e priorizar textos de reclamações de usuários dentro do segmento específico de atendimento ao consumidor, utilizando para tanto técnicas de processamento de texto e aprendizagem de máquina.

Por conta da natureza do trabalho a ser realizado podemos subdividir o esforço necessário para sua conclusão nas seguintes tarefas:

- Imersão na bibliografia
- Desenvolvimento de um módulo de extração de dados de websites contendo as informações textuais das reclamações protocoladas pelos consumidores.

- Formação de uma base de dados única com todas as informações extraídas e devidamente etiquetada por especialistas no domínio de atendimento ao consumidor.
- Aplicação de técnicas de Processamento de Linguagem Natural (PLN) para tratar os dados textuais brutos obtidos.
- Aplicação de técnicas de Aprendizagem de Máquina (AM) para realizar a classificação dos textos.
- Aplicação de técnicas para ranquear os textos previamente classificados como mais relevantes.
- Comparação das técnicas utilizadas e discussão dos resultados obtidos.

1.2 ESTRUTURA DO PROJETO

Nesta seção iremos fazer uma breve explanação sobre a estrutura deste trabalho. Podemos afirmar que o mesmo é composto por 5 capítulos ao todo, incluindo o capítulo atual de introdução.

No capítulo 2, temos a seção denominada áreas correlatas, que visa trazer uma base teórica para que o leitor mesmo não sendo especialista no assunto em questão, consiga compreender ainda que superficialmente, do que se trata esse trabalho.

O capítulo 3 apresenta alguns trabalhos relacionados feitos na área de serviço de atendimento ao consumidor, e que de alguma forma, serviram de inspiração para o desenvolvimento desse projeto.

Já o capítulo 4 trata da estrutura e da implementação do pipeline responsável pelas tarefas de extração, classificação e ranqueamento, mostrando detalhadamente cada um dos passos do pipeline assim como os recursos usados para sua implementação - por exemplo, os algoritmos de aprendizagem de máquina disponíveis. Neste capítulo também apresentamos os experimentos realizados para

avaliação dos algoritmos de aprendizagem de máquina, assim como os resultados obtidos

No capítulo 5 são apresentadas as principais contribuições apresentadas e apontamos as direções para possíveis trabalhos a serem realizados no futuro.

2 ÁREAS CORRELATAS

Neste capítulo apresentaremos conceitos considerados importantes para possibilitar um melhor entendimento do conteúdo abordado no presente trabalho. Na seção 2.1 apresentamos a definição de mineração de texto e detalhamos algumas das tarefas mais relevantes no contexto deste trabalho. A seção 2.2 trata do problema de classificação de texto com ênfase na abordagem baseada em aprendizagem de máquina. Já na seção 2.3 temos alguns conceitos relacionados às métricas de avaliação de resultados. Por fim, na seção 2.4 temos as considerações finais sobre o capítulo.

2.1 MINERAÇÃO DE TEXTO

Mineração de Texto (MT) ou Descoberta de conhecimento em textos (*Knowledge Discovery from Text - KDT*), busca extrair informações úteis a partir da descoberta de padrões e exploração de padrões implícitos numa determinada fonte de dados. Essa atividade se diferencia da mineração de dados tradicional por lidar com um conjunto de documentos de textos não-estruturados, em vez de lidar com uma base de dados normalizada (FELDMAN; SANGER, 2007).

A MT utiliza técnicas de áreas diversas, como recuperação de informação, extração de informação e processamento de linguagem natural em conjunto com métodos de mineração de dados, descoberta de conhecimento em bases de dados (*Knowledge Discovery in Databases - KDD*), aprendizagem de máquina e estatística (HOTHO; NÜRNBERGER; PAAß, 2005).

Recuperação de informação consiste em encontrar informações, normalmente um documento, em um conjunto não-estruturado de documentos, facilitando o acesso à informação (ALLAHYARI, 2017).

Extração de informação refere-se a extração automática de informações relevantes para uma dada tarefa, sobre fontes não-estruturadas ou semi-estruturadas. (MOENS, 2006; SAWARAGI, 2008).

Processamento de Linguagem Natural (PLN) é uma área de pesquisa e aplicações que busca explorar a forma na qual computadores são capazes de entender e manipular textos em forma de linguagem natural. A área de PLN é multidisciplinar, interagindo com diversas áreas adjacentes que proporcionam os

fundamentos para o conhecimento produzido na mesma. Como principais expoentes desses fundamentos temos: Matemática, Ciência da Computação, Linguística e Inteligência Artificial (CHOWDHURY, 2005). Por isso, temos uma outra possível definição para PLN, como sendo um conjunto de técnicas computacionais de diversas áreas com o objetivo de analisar e representar linguagens humanas de forma automática (CHOWDHARY, 2020).

Além das áreas correlatas à mineração de texto mencionadas anteriormente, por conta da abrangência das pesquisas, diversas outras tarefas também podem ser realizadas de acordo com o objetivo da mineração, como, por exemplo: sumarização de textos, clusterização de documentos similares, análise de sentimento entre outros (HOTHO; NÜRNBERGER; PAAß, 2005; ALLAHYARI, 2017).

É importante mencionar que podemos ter algumas interpretações distintas para o termo mineração de texto, de acordo com o enfoque adotado. Alguns autores consideram a mineração de texto quase como um sinônimo de extração de informação, enquanto outros tendem a utilizar o termo quase como um sinônimo para o processo de descoberta de conhecimento em bases de dados (HOTHO; NÜRNBERGER; PAAß, 2005).

Nas seções seguintes, serão apresentadas e detalhadas as tarefas consideradas mais relevantes para proporcionar um melhor entendimento sobre o trabalho desenvolvido.

2.1.1 Coleta de Dados e Formação de Corpus

Existem diversas definições extremamente precisas advindas do campo da linguística que podem ser adotadas para o termo Corpus. Porém a grande maioria delas possuem em características comuns como ser um conjunto de documentos de texto, com tamanho finito e que esse conjunto possa ser reutilizado posteriormente por outros pesquisadores (ALUÍSIO; DE BARCELLOS ALMEIDA, 2006).

Quanto a criação de novos corpus, podemos subdividir essa tarefa em basicamente três etapas:

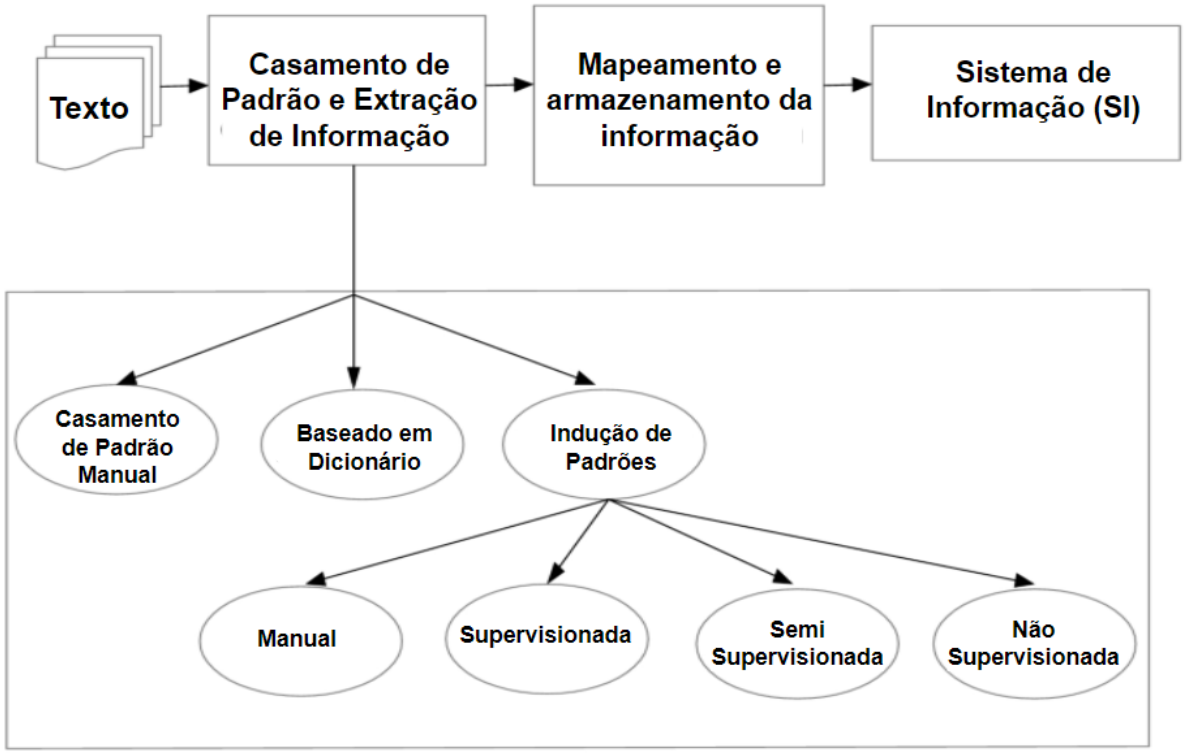
1) Projeto: Consiste na seleção dos textos que serão pertinentes e relevantes
2) Compilação: Trata-se da etapa em que os textos são obtidos, extraídos ou capturados.

3) Anotação: Etapa na qual etiquetamos os textos obtidos anteriormente.

2.1.2 Extração de Informação

A extração de informação tem por objetivo obter informações específicas a partir de determinados documentos de texto, de forma a preencher um template. Essas informações coletadas de forma estruturada, podem ser diretamente exibidas para o usuário, armazenadas num banco de dados para uso posterior ou até mesmo servir de entrada para outros sistemas (ALLAHYARI, 2017; HOTHO; NÜRNBERGER; PAAß, 2005).

Figura 1 - Extração de Informação



Fonte: Adaptado (MIROŃCZUK, 2020)

As diversas abordagens que podem ser utilizadas para realizar a tarefa de extração de informação podem ser subdivididas em três categorias principais: Baseada em casamento de padrões, baseada em dicionário e baseada em aprendizagem de máquina (SINGH, 2018).

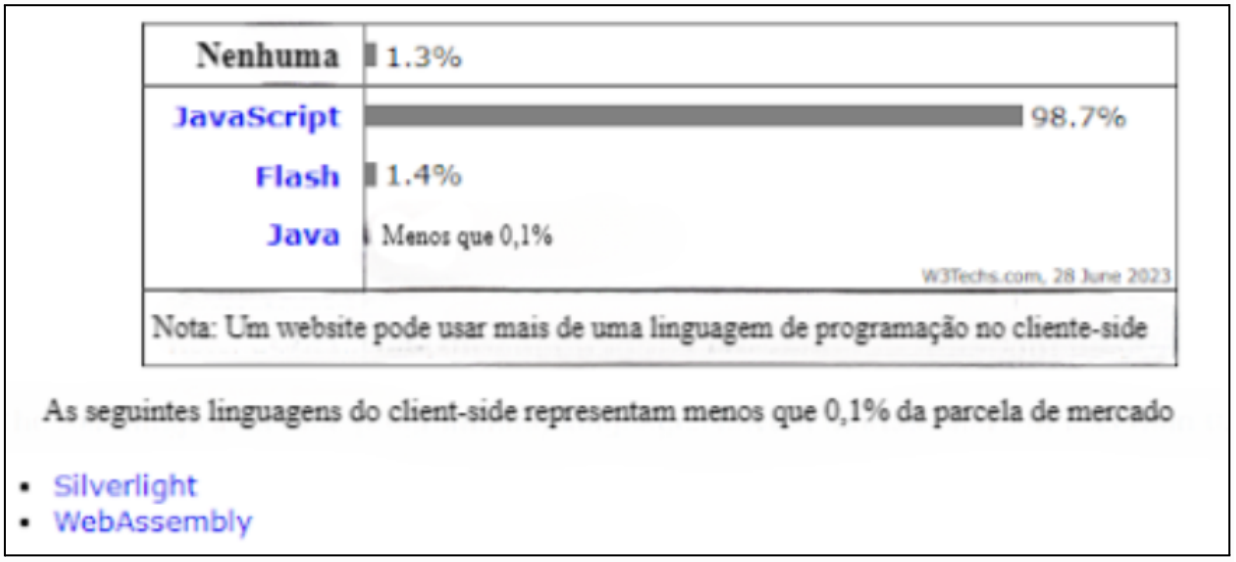
- Casamento de Padrões: Nessa abordagem, padrões de extração são definidos a partir de expressões regulares. Esses padrões são comparados a um determinado texto de entrada e as ocorrências então encontradas, são

extraídas. Apesar de ser um processo simples e rápido que possui algumas limitações, é uma abordagem amplamente utilizada na prática. Vale mencionar que essa abordagem pode ser melhorada a partir de conhecimento especialista no domínio, trazendo consigo informações léxicas e casos especiais (SINGH, 2018).

- Baseada em Dicionário: Nessa abordagem, também conhecida por Gazetteer, utilizamos uma lista pré-definida com todos os possíveis valores que uma determinada entidade a ser extraída pode possuir. Possui uma limitação clara, uma vez que é fortemente dependente da qualidade da lista utilizada (SINGH, 2018; MIROŃCZUK, 2020).
- Baseada em aprendizagem de máquina: Essa abordagem pode ser subdividida em três tipos de acordo com o algoritmo de aprendizagem de máquina utilizado: Supervisionada, Semi-supervisionada ou não-supervisionada. Quando optamos por utilizar algoritmos de natureza supervisionada, utilizamos um conjunto de documentos contendo exemplos que são utilizados durante o processo de aprendizagem. Nesse caso, o conjunto de documentos utilizado no processo de treinamento tem suas entidades de interesse etiquetadas de forma completamente manual. Enquanto na abordagem semi-supervisionada, o conjunto de documentos utilizado para o treinamento é anotado apenas parcialmente de forma manual, tendo uma outra parcela sendo anotada de forma automática. Por fim, no caso de utilizar algoritmos de aprendizagem não-supervisionados, nos utilizamos de métodos para formar clusters contendo entidades similares a serem extraídas (SINGH, 2018; MIROŃCZUK, 2020).

A extração de informação vêm sendo aplicada numa grande diversidade de cenários voltadas para a web. De acordo com a evolução das tecnologias de desenvolvimento para web, a maioria dos grandes web sites começaram a mudar sua estrutura, bem como sua forma de construção especialmente no que diz respeito estratégias de renderização do frontend, utilizando muitos formulários e linguagens de script (SARAWAGI, 2008).

Figura 2 - Uso de linguagens no cliente-side



Fonte: Adaptado (W3Techs, 2023)

Dessa forma, para realizar uma extração de informação textual presente em páginas da web, temos algumas peculiaridades a serem consideradas no acesso a página, como por exemplo, analisar como se dá o processo de renderização das páginas com dados a serem extraídos, se são páginas retornadas estaticamente ou renderizadas de forma dinâmica. Caso a página seja renderizada de forma dinâmica, as duas principais abordagens para lidar com este desafio e conseguir acesso a todas as informações disponíveis na página completa são: Aplicar engenharia reversa na página desejada ou usar alguma ferramenta que simula ou utilize de fato o motor de renderização de um browser (LAWSON, 2015). Esse fator, somado ao conteúdo das páginas em si, são extremamente relevantes para definir que tipo de abordagem e quais técnicas específicas serão utilizadas para realizar a extração da informação, já que temos um espectro bastante amplo, indo desde casamento de padrões, análise de tags html e seletores css, uso expressões regulares até técnicas que se utilizam de aprendizagem de máquina e informações semânticas específicas.

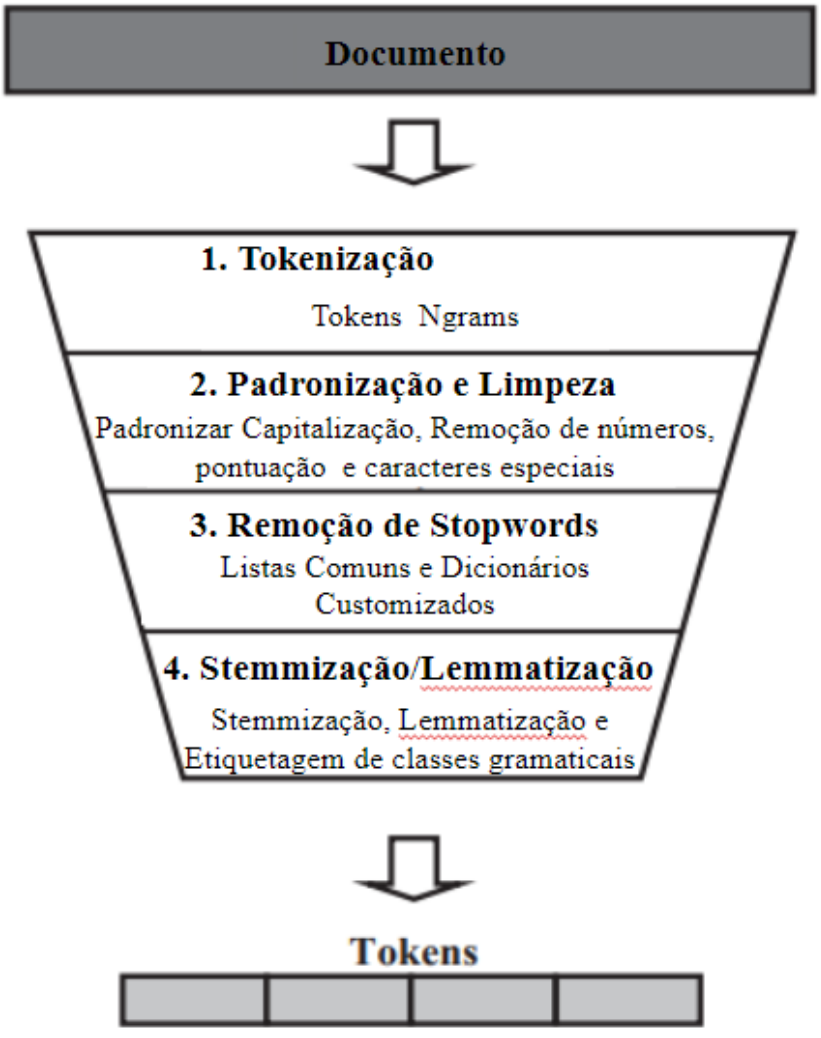
É importante mencionar que as técnicas de extração também variam bastante de acordo com o domínio dos textos em questão. De forma geral, domínios bem limitados possibilitam que a extração ocorra com alta acurácia, enquanto domínios mais genéricos dificultam bastante esse processo. Dessa forma, quando lidamos com domínios genéricos e amplos, muitas vezes se torna necessário o uso de

técnicas e modelos linguísticos complexos para alcançar resultados satisfatórios (RUSSELL; NORVIG, 2002).

2.1.3 Pré-Processamento

Antes de realizar a tarefa de classificação textual, bem como outras atividades relacionadas ao processamento de textos, normalmente efetuamos uma etapa prévia chamada de pré-processamento textual. Essa etapa normalmente consiste em receber como entrada um texto bruto e retorna um conjunto de tokens processados (ANANDARAJAN; HILL; NOLAN, 2019). costuma incluir tarefas como Tokenização, Remoção de StopWords, Conversão para Lowercase e Stemming (UYSAL & GUNAL, 2014).

Figura 3 - Etapas Pré-Processamento Textual



Fonte: Adaptado (ANANDARAJAN; HILL; NOLAN, 2019)

Apesar de que em muitos casos todas as tarefas mostradas na figura 3 sejam executadas em conjunto, de forma sequencial, como partes integrantes da etapa de pré-processamento, não necessariamente o uso de todas elas em conjunto vai trazer os melhores resultados em termos de acurácia ou quaisquer outras métricas definidas para avaliação de performance. Por conta disso, é recomendável testar exaustivamente as diversas combinações de pré-processamentos possíveis e verificar qual dessas combinações é capaz de trazer os melhores resultados de acordo com as métricas de interesse (UYSAL & GUNAL, 2014).

2.1.3.1 Tokenização

Comumente é a primeira etapa de pré-processamento a ser realizada. Consiste em quebrar um determinado texto em partes únicas denominadas tokens. Dependendo da aplicação, nessa etapa certos caracteres indesejados podem ser ignorados, como por exemplo, caracteres especiais e relacionados a pontuação (ALLAHYARI et al, 2017). O Quadro 1 traz um exemplo de tokenização de uma frase de um consumidor insatisfeito com algum produto adquirido.

Quadro 1 - Tokenização

Texto Completo	Estou bastante insatisfeito com o produto de vocês.
Tokens Obtidos	[‘Estou’ , ‘bastante’, ‘insatisfeito’, ‘com’, ‘o’, ‘produto’, ‘de’, ‘vocês’, ‘.’]

Fonte: Autor

Essa etapa é essencial para uma aplicação que envolve PLN. Uma vez que a lista de tokens retornada no processo de tokenização, serve como entrada para a maioria das etapas de pré-processamento que possam vir a ser utilizadas posteriormente.

Apesar de se uma das etapas consideradas mais simples, ela pode se tornar um pouco mais complexa em casos específicos, como por exemplo, tokenizadores que lidam com linguagens de marcação como HTML, SGML ou XML (RUSSELL;

NORVIG, 2002) ou que lidam com N-Grams que basicamente são tokens compostos por N palavras consecutivas (ANANDARAJAN; HILL; NOLAN, 2019).

2.1.3.2 Remoção de Stopwords

Uma tarefa clássica utilizada em grande parte dos sistemas que envolvem PLN e recuperação de informação. Consiste em remover do texto palavras que aparecem com alta frequência e que não trazem muita informação (conteúdo semântico). Essas palavras são incluídas na *stoplist,* uma lista de palavras que serão removidas. Como exemplos principais temos palavras cuja função principal seja de conectar termos ou orações, ou seja, palavras de classes gramaticais como preposições e conjunções. (ALLAHYARI et al, 2017). Além disso, também podemos ter artigos, numerais e alguns pronomes fazendo parte da *stoplist.* Em casos especiais, substantivos muito frequentes sem poder discriminatórios podem ser incluídos.

Figura 4 - Exemplos de Stopwords

Lista de Stopwords		
de	e	a
que	para	o
com	em	isso

Fonte: Autor

É importante mencionar que Stopwords são palavras específicas e que variam de acordo com a linguagem e suas regras de construção. Apesar de muitas vezes serem consideradas irrelevantes e até prejudiciais sendo portanto removidas, existem cenários em determinados domínios e linguagens nos quais as Stopwords possuem informação relevante e sua remoção acaba resultando em um queda no

desempenho numa posterior tarefa de classificação textual (UYSAL & GUNAL, 2014).

Por esses fatores é sempre válido verificar se a remoção das Stopwords é realmente uma tarefa que realmente vale a pena ser realizada. Essa decisão irá variar de acordo com a linguagem a ser trabalhada, com o domínio da aplicação, bem como com a natureza das métricas a serem utilizadas.

2.1.3.3 Stemmização

Essa tarefa busca simplificar um determinado token obtendo sempre a sua raiz em detrimento de sua derivação. Uma vez que palavras derivadas são normalmente semanticamente similares à sua forma raiz, essa tarefa é bastante utilizada para terminar a ocorrência de terminado termos no texto (UYSAL & GUNAL, 2014). O Quadro 2, a seguir, traz exemplos simples de stemmização.

Quadro 2 - Exemplos de Stemming

Palavra	Palavra após aplicar Stemming
aprender	aprend
carreira	carr
reclamação	reclam

Fonte: Autor

É válido destacar que essa é uma tarefa bastante dependente da língua adotada, uma vez que sua própria relevância varia de acordo com a complexidade morfológica do idioma. Inclusive os próprios algoritmos utilizados para realização da tarefa variam de acordo com a língua adotada. Porém de forma mais genérica podemos classificar os algoritmos de Stemming em três abordagens principais: Truncagem, Estatísticas e Mista (VIJAYARANI et al, 2015).

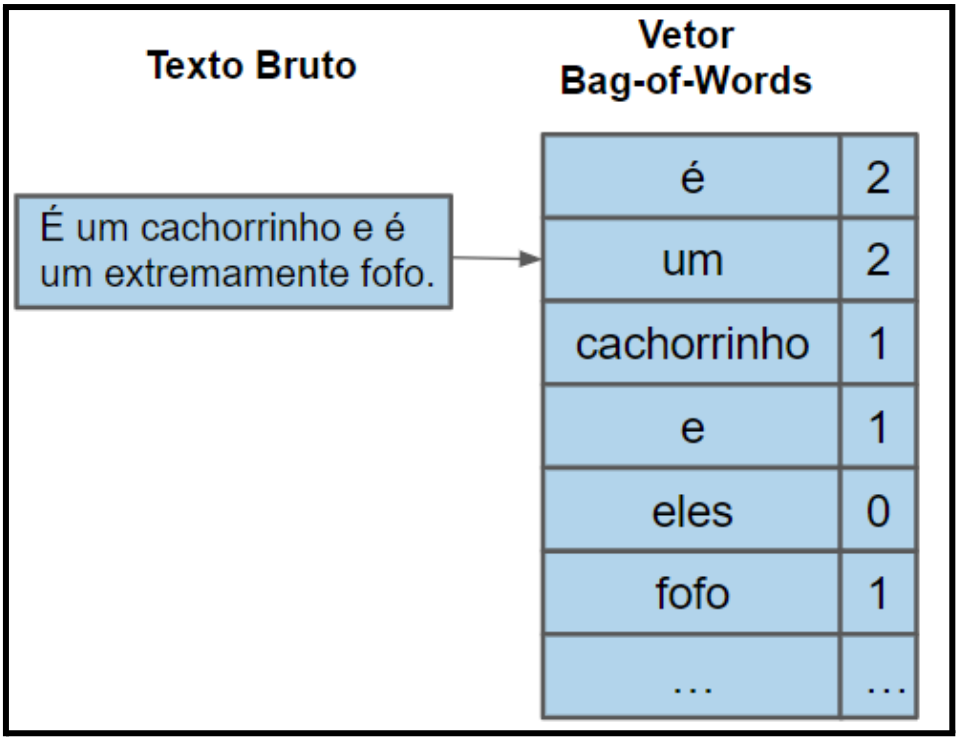
2.1.4 Transformação do Texto

Algoritmos de aprendizagem de máquina normalmente não são capazes de lidar com textos brutos. Eles lidam melhor com representações numéricas. Por isso

se torna necessária a utilização de técnicas capazes de realizar essa tarefa de transformação. Dentre as mais usadas atualmente temos “Bag-Of-Words” e “TF-IDF”.

Na abordagem bag-of-words o texto em questão é transformado em um vetor contendo as ocorrências (frequência) de cada uma das palavras. É considerada uma abordagem bastante simples, porém ainda sim efetiva. Sendo uma das mais utilizadas até hoje por essas características.

Figura 5 - Exemplo da abordagem Bag-of-Words



Fonte: Adaptado (ZHENG; CASARI, 2018)

Por padrão, essa abordagem não possui a noção de sequência, dessa forma, a ordem em que as palavras aparecem no vetor não é considerada, apenas a quantidade de vezes que elas aparecem no texto em questão. Outra dificuldade que pode ocorrer ao utilizar essa abordagem consiste na quebra de frases em palavras únicas. Uma vez que existem palavras que quando combinadas passam uma noção semântica distinta ou até mesmo oposta ao sentido das palavras consideradas individualmente (ZHENG; CASARI, 2018). Como exemplo podemos mencionar “nada bem”.

TF-IDF é uma abreviação que vem do inglês “Term-Frequency – Inverse Document Frequency”. Consiste num método estatístico que mede a importância de uma palavra em um conjunto de documentos. Ela é calculada através da multiplicação de duas métricas: O total de vezes que uma determinada palavra aparece num documento (TF) e o inverso da frequência das palavras num documento (IDF) (AKUMA; LUBEM; ADOM, 2022). Na prática essa abordagem pode ser considerada como um simples ajuste sobre a abordagem Bag-of-Words. Utilizamos a quantidade de palavras normalizada, isto é, dividimos a quantidade de ocorrências de uma determinada palavra pela quantidade de documentos nas quais elas aparecem em vez da quantidade de ocorrências bruta que ocorre na abordagem BoW (ZHENG; CASARI, 2018). Esse simples ajuste permite que palavras que apareçam menos vezes nos documentos, mas que normalmente carregam maior valor semântico, possuam uma maior valorização em termos de importância.

2.2 CLASSIFICAÇÃO DE TEXTO BASEADA EM APRENDIZAGEM DE MÁQUINA

A classificação textual também conhecida como categorização, consiste em uma tarefa na qual, dado um determinado documento de texto, decidimos a qual classe esse documento pertence, dado um conjunto predefinido de classes ou categorias possíveis (RUSSELL; NORVIG, 2002; IKONOMAKIS; KOTSIANTIS; TAMPAKAS, 2005).

Essa tarefa possui um grande número de aplicações no nosso cotidiano, como por exemplo, identificação automática da linguagem, classificação de gênero, detecção de spam entre outros (RUSSELL; NORVIG, 2002).

Essa tarefa costuma ser realizada utilizando uma das duas abordagens a seguir: engenharia do conhecimento, que se baseia em um conjunto de regras definidas por especialistas ou por aprendizagem de máquina. Considerando o escopo do nosso trabalho, vamos discutir aqui apenas a abordagem baseada em aprendizagem de máquina.

Uma das definições clássicas de aprendizagem de máquina diz que um determinado programa é capaz de aprender sobre uma determinada experiência “E” sobre uma tarefa “T”, cuja performance pode ser medida por “P”, se a performance

na tarefa “T”, que é medida por “P” melhora de acordo com a experiência “E” (MITCHELL, 1997). Outra definição de natureza semelhante diz que o conceito de aprendizagem é baseado no princípio de treinar as máquinas computacionais e possibilitar que elas aprendam por si só (CHOWDHARY, 2020).

Quanto às abordagens, podemos subdividir a área de AM em 3 tipos principais de aprendizagem: Supervisionado, Não Supervisionado e Aprendizado por Reforço (RUSSELL; NORVIG, 2002).

No aprendizado supervisionado cada um dos elementos do conjunto do treinamento possui uma etiqueta (HAN; PEI; KAMBER, 2011). Essa etiqueta representa a definição da classe cujo elemento em questão faz parte.

Já no aprendizado não supervisionado, essa informação extra não é conhecida, as classes dos elementos do conjunto de treinamento não são identificadas. Dessa forma, os algoritmos com esse tipo de abordagem tentam agrupar os elementos que sejam mais prováveis de pertencer a uma mesma classe ou grupo (HAN; PEI; KAMBER, 2011).

O aprendizado por reforço é uma abordagem que baseia o aprendizado numa série de reforços, que consistem em recompensas e punições que vão sendo distribuídas de acordo com as ações realizadas previamente (RUSSELL; NORVIG, 2002). Os algoritmos deste tipo de aprendizado são utilizados principalmente em problemas que expressam uma noção de constância, cujas ações ao longo do tempo irão determinar os reforços que serão recebidos e impactarão nas prováveis soluções.

Por conta do escopo deste trabalho, temos um maior enfoque nos algoritmos de classificação baseados na abordagem de aprendizado supervisionado.

2.2.1 Algoritmos de Classificação (Aprendizagem Supervisionada)

Nesta seção apresentaremos alguns dos principais algoritmos com a abordagem de aprendizagem supervisionada que são utilizados para a tarefa de classificação.

Apesar de todos seguirem a abordagem supervisionada, eles podem ter origens, inspirações e técnicas de construção bastante distintas. Conforme veremos a seguir.

2.2.1.1 Regressão Logística

É um algoritmo de classificação supervisionado que se baseia em métodos estatísticos. A regressão logística basicamente propõe um mecanismo que permite aplicar a técnica de regressão linear para problemas de classificação (SAMMUT; WEBB, 2011). Ela funciona de maneira bastante semelhante à regressão linear, se diferenciando por ter uma resposta binária como saída (SPERANDEI, 2014). Essa resposta também é comumente chamada de variável dependente.

$$Y = \alpha + \beta x \qquad \text{(Eq. 1)}$$

Temos que "Y" seria a variável dependente que é calculada em função de "x" que é considerada a variável independente. A regressão logística pode ser chamada de binária ou univariada quando temos apenas uma variável independente. Já caso a função possua mais de uma variável independente, temos uma regressão logística múltipla.

$$Y = \alpha + \beta 1X1 + \beta 2X2 + \beta kXk \qquad \text{(Eq. 2)}$$

Ao aplicarmos a função *logit*, à equação de uma das funções lineares descritas nas equações anteriores, temos a fórmula do modelo logístico que é dada por:

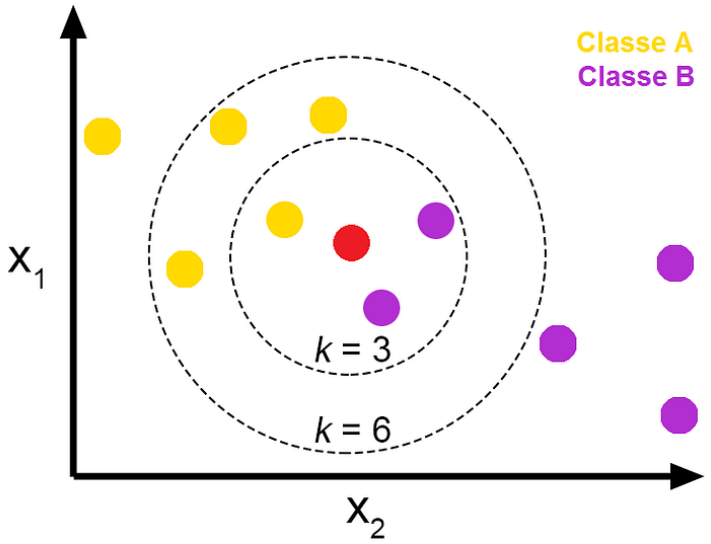
$$p(x) = \frac{1}{1 + e^{-(\alpha+\beta 1X1)}} \qquad \text{(Eq. 3)}$$

Possui uma característica particularmente interessante se comparado com outros algoritmos semelhantes, a capacidade de retornar as probabilidades associadas à predição. Podendo inclusive utilizá-las sob outro modelo ou algoritmo de aprendizagem para identificar as probabilidades associadas à classe alvo da predição (HILBE, 2015).

2.2.1.2 K-Nearest Neighbors (KNN)

É um algoritmo de classificação não-paramétrico cuja abordagem de aprendizagem pode ser categorizada como baseada em instâncias (RUSSELL; NORVIG, 2002). O algoritmo assume que todas as instâncias correspondem a pontos num espaço n-dimensional (MITCHELL, 1997) . Dessa forma, a classificação é realizada de acordo com as "K" instâncias mais próximas do ponto a ser classificado.

Figura 6 - Escolha do parâmetro "K" no classificador KNN



Fonte: (JOSÉ, 2018)

Já o conceito de proximidade entre dois pontos pode ser definido através de modelos já bastante consolidados, como por exemplo: a distância euclidiana padrão, a distância de Manhattan e a distância de Minkowski.

Figura 7 - Tipos de distâncias comumente usadas para o KNN

$$Manhattan\ Distance = \sum_{i=1}^{d} |x_{1i} - x_{2i}|$$

$$Euclidean\ Distance = \left(\sum_{i=1}^{d} (x_{1i} - x_{2i})^2 \right)^{\frac{1}{2}}$$

$$Minkowski\ Distance = \left(\sum_{i=1}^{d} |x_{1i} - x_{2i}|^p \right)^{\frac{1}{p}}$$

Fonte:(TADAGOPPULA, 2020). Adaptada

A escolha do parâmetro “K” é considerada extremamente importante para a execução desse algoritmo (MOLDAGULOVA; SULAIMAN, 2017). Uma vez que valores muito pequenos do parâmetro tendem a resultar em overfitting, enquanto

valores de “K” muito elevados causam underfitting. Por isso existem diversas técnicas específicas para a escolha de um valor satisfatório para parâmetro de entrada “K”.

2.2.1.3 Naive Bayes

É um algoritmo de aprendizagem supervisionado da família dos classificadores Bayesianos que são por natureza classificadores estatísticos. A classificação é realizada com base no teorema de Bayes e usualmente possuem alta acurácia e boa performance de tempo de execução quando aplicados em base de dados grandes (HAN; PEI; KAMBER, 2011).

Naive Bayes é a alcunha dada a um classificador Bayesiano simples, pois assume que o efeito do valor de um determinado atributo de uma classe é independente dos valores assumidos por outros atributos (HAN; PEI; KAMBER, 2011). Esta suposição, é chamada de independência condicional, e por conta dessa característica única, temos a denominação “Naive”, traduzindo literalmente “Ingênuo”.

Dado que “x” representa um vetor com valores dos atributos utilizados no treinamento e “y” as classes de interesse. A classe “C” atribuída por esse classificador é dada através da seguinte equação:

$$C = \text{argmax}_{Yi} \prod_{j} P(x_j|y_i)\, P(y_i) \qquad \text{(Eq. 4)}$$

Apesar da presunção de que os atributos utilizados no processo de aprendizagem serem condicionalmente independentes ser frequentemente desrespeitada, ainda sim esse algoritmo desempenha de forma competitiva frente a diversos outros classificadores. É considerado eficiente uma vez que o treinamento

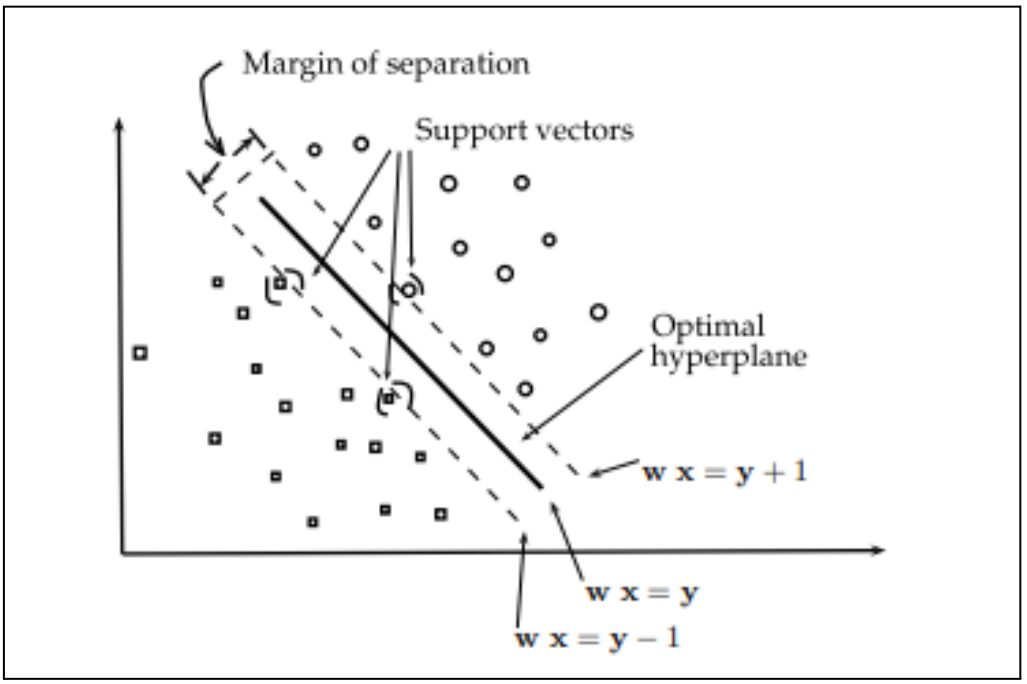
e classificação são realizadas em tempo linear. Além disso, também possui uma construção incremental, os dados de treinamento podem ser facilmente atualizados uma vez que novos dados vão sendo adquiridos. Essas características fazem com que esse classificador seja bastante utilizado na prática (WEBB; KEOGH; MIIKKULAINEN, 2010; CHOWDHARY, 2020).

2.2.1.4 Support Vector Machine (SVM)

É um dos algoritmos de aprendizagem supervisionada mais comuns atualmente, sendo quase sempre uma escolha interessante quando ainda não temos um conhecimento aprofundado da relação dos algoritmos de aprendizagem de máquina com o domínio em questão (RUSSELL; NORVIG, 2002).

Basicamente esse algoritmo tenta buscar a reta ou hiperplano que melhor separa as classes alvo e complementar. É importante mencionar que por possuir características robustas, como a capacidade de transformação dos dados para uma dimensão superior, o algoritmo é capaz de lidar com classificação tanto com dados lineares quanto com não-lineares (RUSSELL; NORVIG, 2002; HAN; PEI; KAMBER, 2011).

Figura 8 - Conceitos utilizados no SVM



Fonte: (CHOWDHARY, 2020)

A margem de separação é a distância entre os limites estabelecidos a partir da separação dos conjuntos contendo os elementos das classes. Ela pode ser considerada uma métrica de segurança na separação entre os conjuntos de pontos, uma vez que quanto maior a margem, maior a separação entre os conjuntos. Um hiperplano considerado ótimo é computado através da maximização da margem de separação (CHOWDHARY, 2020).

$$W \bullet X + b = 0 \qquad \text{(Eq. 5)}$$

A equação teta descreve um hiperplano de separação ótimo. Nessa equação, temos que “W” representa um vetor de pesos, “X” representa um vetor contendo os pontos do conjunto de treinamento e “b” sendo um escalar comumente chamado de bias (HAN; PEI; KAMBER, 2011; CHOWDHARY, 2020).

A abordagem de transformar os dados de treinamento originais em dados com um maior número de dimensões é realizada quando não é possível encontrar um hiperplano de separação linear dentro da dimensionalidade padrão. Para tanto é realizado um mapeamento não linear dos dados de treinamento através de uma função Kernel. Essa abordagem acaba sendo importante pois frequentemente temos dados não linearmente separáveis no espaço dimensional original, porém que podem ser facilmente separáveis linearmente em um espaço dimensional superior. De forma que o separador linear numa dimensão superior, é na verdade não-linear no espaço original (RUSSELL; NORVIG, 2002; HAN; PEI; KAMBER, 2011).

Existem algumas funções Kernel que são bastante utilizadas, entre elas temos: Polinomial, Gaussianas, Sigmoid e RBF (Radial Basis Function). É importante mencionar que não existem regras explícitas sobre qual dessas funções teria os melhores resultados durante a execução do classificador. Inclusive na prática, em diversos casos, a escolha do Kernel tende a não causar uma diferença tão significativa em termos de acurácia (HAN; PEI; KAMBER, 2011).

2.2.1.5 Random Forest

Um classificador Random Forest consiste basicamente em um conjunto de classificadores independentes baseados em árvore de decisão em que cada um

desse classificadores possui um voto único e a classe resultante é produto de uma simples maioria dos votos (BREIMAN, 2001). Esse algoritmo de aprendizagem de máquina supervisionada tem entre suas características rodar relativamente rápido, produzir bons resultados em termos de acurácia, costuma lidar bem com dados que contém uma quantidade relevante de ruídos, além de não ser tão suscetível à overfitting (LIU; WANG; ZHANG, 2012).

O resultado final desse classificador pode ser dado pela seguinte fórmula:

$$H(x) = \arg\max \sum_{i=1}^{k} I\,(\,h_i(x)=Y\,) \qquad \text{(Eq. 6)}$$

Na equação, temos que H(x) é o resultado do classificador, h_i representa um único modelo de classificação baseado em árvore de decisão, Y é a variável de saída e I é a função indicadora.

Algoritmos de classificação baseados em árvore de decisão são bastante utilizados por poderem ser facilmente representados por um conjunto de regras “se-então”, de forma que seus resultados podem ser facilmente interpretados inclusive por pessoas com pouca experiência na área (RUSSEL; NORVIG,).

Ensemble é uma técnica de aprendizagem de máquina que consiste em aumentar o espaço de hipóteses através da combinação dos resultados de diferentes predições realizadas sobre um mesmo conjunto, com objetivo de minimizar a quantidade de erros de predição. (RUSSELL; NORVIG). O algoritmo Random Forest utiliza-se de um tipo específico de Ensemble denominado Bagging no qual cada um dos classificadores baseados em árvore de decisão é treinado a partir de diferentes pedaços do conjunto de treinamento e cujas predições são combinadas no final, a partir de uma votação (SAMMUT; WEBB, 2011).

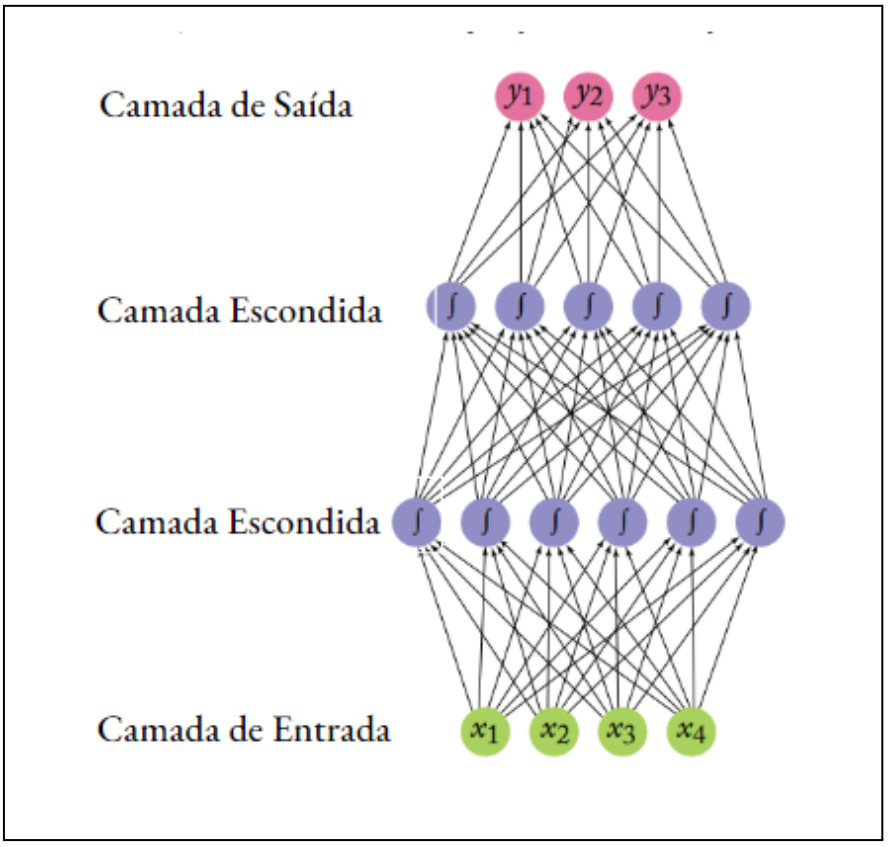
2.2.1.6 MultiLayer Perceptron (MLP)

O estudo das redes neurais artificiais (RNAs) foi inspirado a partir da observação do sistema de aprendizado biológico que é construído a partir de um conjunto de neurônios interconectados. De forma análoga, nas redes neurais

artificiais nos utilizamos de um conjunto de unidades simples (que representam os neurônios) conectadas entre si que recebem uma entrada e produzem uma saída (MITCHELL, 1997).

Um classificador MLP é uma rede neural feedforward na qual a informação é transmitida de forma unidirecional da camada de entrada até a camada de saída, passando por ao menos uma camada interna escondida (BISHOP, 1995).

Figura 9 - Rede Neural Feed-Forward com duas camadas escondidas de 5 e 6 neurônios



Fonte: Adaptado (GOLDBERG, 2017)

As redes neurais feed forwards são aquelas de sentido único, ou seja, não contém loops e portanto, sua saída pode ser calculada explicitamente em função da entrada, dos pesos e dos biases (BISHOP, 1995).

Observando a figura 9, temos que cada círculo representa um neurônio e cada seta representa suas entradas e saídas de acordo com seu sentido. É importante mencionar que cada uma dessas setas carrega consigo pesos associados. Trazendo para uma notação matemática, uma rede neural como a da figura 9, poderia ser representada da seguinte forma:

$$R = (g^2(g^1(xW^1 + b^1)\,W^2 + b^2))\,W^3 \qquad \text{(Eq. 7)}$$

Onde nesta equação, temos que x é a entrada e W1, b1 e g1 são respectivamente uma matriz de pesos, o bias e a função de ativação da primeira camada escondida, enquanto W2, b2 e g2 possuem as mesmas funções, porém como parte da segunda camada escondida.

Cada “conjunto” xW1 + b1 é na verdade uma transformação linear, sendo que cada uma dessas transformações lineares representa uma camada. Ou seja, como uma transformação linear sobre outra, continua sendo uma transformação linear, podemos adicionar quantas camadas desejarmos. RNAs com várias camadas escondidas, são consideradas profundas, daí o termo aprendizagem profunda ou *deep learning* (GOLDBERG, 2017).

2.3 AVALIAÇÃO DE RESULTADOS

Quando lidamos com algoritmos de aprendizagem máquina, as seguintes métricas são comumente utilizadas durante seu processo de avaliação: Precisão, Cobertura, Acurácia e F-Measure (HAN; PEI; KAMBER, 2011).

Antes de definir as métricas citadas anteriormente, é válido se familiarizar com as terminologias apresentadas no quadro abaixo, uma vez que serão utilizadas na apresentação dos conceitos das métricas utilizadas para avaliação.

Quadro 3 - Terminologias usadas no cálculo das métricas

Verdadeiros Positivos (VP)	Ocorre quando um texto da classe alvo é atribuído corretamente pelo classificador
Verdadeiros Negativos (VN)	Ocorre quando um texto da classe complementar é atribuído corretamente pelo classificador.
Falsos Positivos (FP)	Ocorre quando um texto da classe complementar é atribuído incorretamente pelo classificador.
Falsos Negativos (FN)	Ocorre quando um texto da classe alvo é atribuído incorretamente pelo classificador

Fonte: Autor

Com isto, podemos utilizar os conceitos apresentados no quadro 3 para definir matematicamente as métricas utilizadas no processo de avaliação dos classificadores utilizados.

A métrica de precisão refere-se à quantos dos elementos classificados como positivos, realmente pertencem a essa classe. Ou seja, uma métrica ligada à exatidão.

$$\text{Precisão} = \frac{VP}{VP+FP} \quad \text{(Eq. 8)}$$

Já a métrica cobertura refere-se à quantos dos elementos que realmente são positivos foram corretamente atribuídos à classe alvo. Ligada à completude.

$$\text{Cobertura} = \frac{VP}{VP+FN} \quad \text{(Eq. 9)}$$

A acurácia refere-se à quantidade dos elementos corretamente classificados, independentemente de sua classe. Sendo uma das mais simples, e mais utilizadas métricas.

$$\text{Acurácia} = \frac{VP+VN}{VP+FP+VN+FN} \quad \text{(Eq. 10)}$$

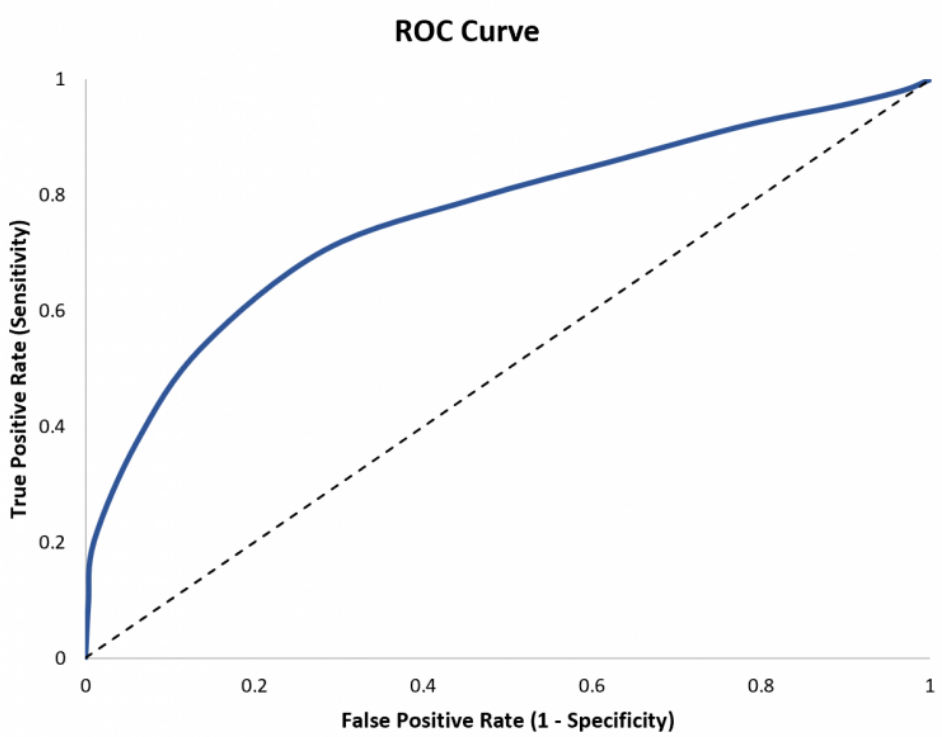
F-Measure trata-se da média harmônica entre precisão e cobertura. É bastante usada por tornar perceptíveis distorções entre os valores obtidos para precisão e cobertura.

$$\text{F-Measure} = 2 \times \frac{Precisão * Cobertura}{Precisão + Cobertura} \quad \text{(Eq. 11)}$$

A curva ROC (Receiver Operating Characteristic) trata do relacionamento entre as taxas dos verdadeiros positivos (cobertura ou sensibilidade) e dos falsos positivos (SAMMUT; WEBB, 2011). Bastante utilizada por possibilitar uma análise visual.

Na figura 10 a seguir, apresentamos um exemplo do gráfico de uma curva ROC, exibindo a taxa dos verdadeiros positivos (sensibilidade) no eixo “Y” e a taxa dos falsos positivos no eixo “X”.

Figura 10 - Exemplo Curva ROC



Fonte: (ZACH, 2021)

A área sob a curva (AUC) é considerada como uma métrica capaz de verificar a qualidade de um modelo percebendo distorções no balanceamento entre as classes classificadas, que passariam despercebidas por métricas como acurácia por exemplo. Quanto mais próximo do quadrante superior esquerdo e portanto com maior área, melhor a qualidade do modelo.

2.4 CONSIDERAÇÕES FINAIS

Este capítulo teve como foco apresentar conceitos básicos e definições das áreas de pesquisa relacionadas ao tema do nosso trabalho: extração de informação e classificação e texto.

O capítulo 3 irá apresentar alguns trabalhos relacionados ao que foi desenvolvido neste trabalho, bem como realizará uma comparação entre os mesmos, buscando esclarecer semelhanças e diferenças.

3 TRABALHOS RELACIONADOS

Neste capítulo apresentaremos alguns dos trabalhos relacionados ao tema desta pesquisa, seja por servirem como fonte de informação sobre o tema ou por utilizarem técnicas semelhantes, como Mineração de texto e Aprendizagem de Máquina para resolver problemas originados ou relacionados à serviços de atendimento ao consumidor. As seções deste capítulo estão organizadas da seguinte forma: Na Seção 3.1 mencionamos diversos trabalhos que servem como base para um mínimo entendimento sobre comportamento de consumidores, serviços de atendimento e respostas dos consumidores à insatisfação. Na Seção 3.2 tratamos de trabalhos relacionados aos serviços de atendimento ao consumidor que lidam com aprendizagem de máquina e dados de natureza categórica ou numérica. Já na Seção 3.3 apresentamos trabalhos que lidam com aprendizagem de máquina e dados de natureza textual. Por fim, na Seção 3.4 apresentamos as considerações finais deste capítulo.

3.1 SERVIÇO DE ATENDIMENTO AO CONSUMIDOR (SAC)

É comum que consumidores façam escolhas de onde consumir, baseados não apenas no produto ou serviço adquirido no momento da compra, mas também que se baseiem na qualidade do atendimento que eles esperam receber após a compra, caso aconteça algum tipo de problema ou dificuldade (BLODGETT; WAKEFIELD; BARNES, 1995).

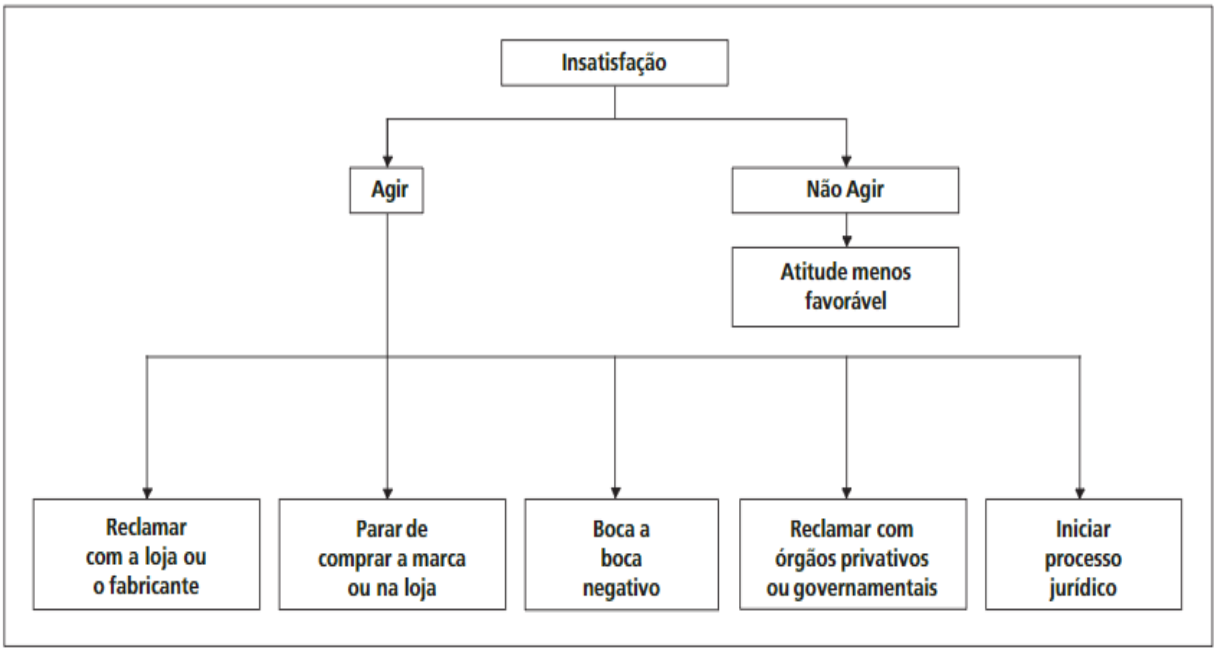
Os serviços de atendimento ao consumidor buscam estabelecer um pipeline de comunicação entre uma determinada empresa e seus respectivos clientes ou potenciais futuros clientes (SCHIESSL, 2007). Eles se popularizaram no Brasil a partir da década de 1990 com a instituição do Código de Defesa do Consumidor (CHAUVEL; GOULART, 2007). Desde então eles vêm se adaptando e modernizando de acordo com as novas tecnologias que estão surgindo e se popularizando, especialmente com o advento da internet.

Apesar da instituição de serviços de atendimento ser algo de grande interesse e benéfico para os consumidores, é importante mencionar que as empresas também possuem motivações próprias para incentivar seus clientes a procurar seus serviços de atendimento em caso de insatisfação, uma vez que alguns clientes insatisfeitos

podem simplesmente não consumir mais nada relacionado a empresa e até mesmo realizar boca-a-boca negativo (inclusive e principalmente a sua versão mais atual, eletrônica) sem aviso prévio. Isso pode acontecer por diversos motivos. seja por conta da dificuldade ou demora em receber atendimento, pela incredulidade de que a raiz de sua insatisfação acabe sendo realmente resolvida ou até mesmo por uma simples relutância em registrar uma reclamação (BLODGETT; WAKEFIELD; BARNES, 1995).

Já Hawkins, Mothersbaugh & Best (2007) propõem que clientes insatisfeitos tendem a tomar uma das seguintes decisões apresentadas na figura a seguir.

Figura 11 - Resposta à Insatisfação



Fonte: (HAWKINS, MOTHERSBAUGH & BEST, 2007)

Considerando as possíveis decisões apresentadas, fica claro que caso um determinado consumidor tome a decisão de agir sobre a sua insatisfação, o melhor cenário existente para a empresa, é justamente receber apenas uma reclamação. Afinal, todas as outras possibilidades de respostas à insatisfação tendem a causar prejuízos consideráveis uma vez que sejam efetivamente tomadas. Enquanto a reclamação, por si só, ainda permite uma oportunidade de comunicação e possibilidade de remediação do problema apresentado.

Por isso, é importante tentar evitar que esses cenários considerados piores ocorram, fazendo o possível para evitar que uma simples reclamação acabe escalando até chegar num ponto que possa resultar em enormes prejuízos para as empresas. Dado que atrair um novo consumidor pode custar cerca de cinco vezes mais do que simplesmente fazer o esforço necessário para conseguir reter um cliente já estabelecido (DESATNICK, 1988).

É válido mencionar que as respostas mais radicais à insatisfação são menos comuns quando lidam com empresas consideradas responsáveis e responsivas às reclamações registradas. Dessa forma, a aplicação de estratégias de comunicação ativas nos seus canais são capazes de influenciar na diminuição dessas respostas mais danosas as empresas (SINGH, 1990).

No contexto atual, um fenômeno bastante interessante vem ocorrendo, o da descentralização do acesso a esses serviços de comunicação. As empresas buscam estar cada vez mais presentes e acessíveis ao consumidor, muitas vezes contam não mais com apenas um ponto central de contato. Mas sim diversas possibilidades, como: Telefone, Site da empresa, Redes sociais como Twitter, Facebook e Instagram. Assim como páginas próprias como Reclame Aqui e Consumidor.gov.

Por conta da explosão no uso das redes sociais, comentários negativos tendem a se espalhar rapidamente através do boca-a-boca eletrônico. Dessa forma, é importante a atenção das empresas no monitoramento sobre comentários e reclamações de carga fortemente negativa com potencial para “viralizar” (KUMAR; REINARTZ, 2018).

Isso resulta em um grande volume de informações, em sua maioria de natureza textual livre, causando grandes dificuldades das empresas para conseguir atender todas essas solicitações de contato em tempo hábil. Nesse contexto surge a necessidade de analisar abordagens e buscar técnicas que possibilitem processar ou organizar essas informações, de forma a agilizar este processo. Seja através de classificações por tipo ou natureza, seja por agrupamentos de solicitações comuns, ou até mesmo utilizando-se de técnicas de ranqueamento, essas são apenas algumas dentre as diversas possibilidades de abordagens que possam vir a ser aplicadas no processamento dessas informações textuais de acordo com as necessidades específicas de cada empresa.

Existem diversos artigos que trabalham com algum tipo de aprendizagem de máquina para resolver problemas ligados ao setor de atendimento ao consumidor. Dentre os quais, optamos por subdividir os trabalhos em duas categorias relacionadas ao tipo de dado que será trabalhado no artigo em questão. Dessa forma, temos uma seção com os trabalhos que apresentam dados de natureza categórica ou numérica e uma outra seção apenas com os de natureza textual.

3.2 DADOS CATEGÓRICOS E NUMÉRICOS

Em (RABBI et al., 2018) aplicam-se técnicas de mineração de dados utilizando a metodologia KDD (Knowledge Discovery in Databases) para obter novos conhecimentos a partir de uma base de dados de reclamações pública disponibilizada pelo Programa de Proteção e Defesa do Consumidor (Procon) que funciona como um órgão auxiliar do Poder Judiciário.

Na etapa de execução da mineração, a mesma é realizada utilizando-se da plataforma WEKA, na qual são executados algoritmos de duas abordagens distintas, o K Means para clusterização dos dados (aprendizagem não-supervisionada) e as árvores de decisão J48 (aprendizagem supervisionada) para obter insights e possíveis regras de negócio implícitas na base de dados.

Apesar de uma temática semelhante ao nosso projeto por possuir foco em reclamações e no atendimento aos consumidores, diferencia-se por não possuir um foco prévio em uma atividade específica, uma vez que seu objetivo é mais genérico, o de identificar padrões e tendências nos dados, sejam eles de natureza categórica ou numérica. Para tanto, foi utilizada apenas uma técnica de aprendizagem de máquina não-supervisionada para categorizar os registros das reclamações baseados em características comuns. Numa etapa posterior, é utilizada também apenas uma única técnica de aprendizagem supervisionada baseada em árvore de decisão para tentar obter regras implícitas sobre quais os atributos que influenciam um dos atributos de maior interesse da pesquisa, o tempo de atendimento.

Já em (GHAZZAWI; ALHARBI, 2019) utilizam-se de dados públicos obtidos pelo setor de relacionamento com o cliente de uma entidade que provê serviços de transporte público na região da América do Norte. São aplicadas técnicas de

mineração de dados para tentar obter informações que possam levar a insights que melhorem a qualidade do serviço ou que aumentem a satisfação do consumidor.

Para tanto, eles analisam as correlações existentes nos atributos do dataset obtido, tentam identificar as principais causas de reclamações além de predizer a agência de origem de um determinado registro utilizando modelos de classificação por aprendizagem de máquina (KNN, Naive Bayes, Random Trees e ID3).

Quanto à avaliação dos resultados da tarefa de classificação, utilizam-se de 10-Fold Cross Validation e métricas como Acurácia, Precisão e Cobertura. O algoritmo ID3 (Iterative Dichotomiser) que é usado para gerar árvores de decisão sendo o precursor do C4.5 foi o que teve melhores resultados.

Diferente de nosso projeto, ele se concentra em classificar as reclamações de acordo com sua respectiva agência de origem, utilizando-se de dados que podem ser considerados de natureza mais categórica ou numérica do que uma análise mais voltada à elementos de natureza textual.

3.3 DADOS TEXTUAIS

Em (VERMEER et al., 2019) busca-se entender o “boca-a-boca” eletrônico sobre um conjunto de empresas. Para tanto são coletados comentários obtidos através do twitter e do facebook da empresa. São realizadas basicamente duas etapas de classificação. Na primeira uma classificação binária se o texto realmente fala sobre algo relacionado a empresa (produto, serviço etc). E somente em caso positivo os textos passam por uma segunda etapa, uma análise de sentimento (positivo, neutro, negativo).

Cerca de 5% dos dados coletados foram etiquetados manualmente de acordo com as duas etapas de classificação mencionadas. É realizada então uma comparação sobre algumas das abordagens mais comuns ao realizar uma análise de sentimento, entre elas estavam a baseada em dicionário, além de diversos algoritmos de aprendizagem de máquina.

De acordo com os resultados apresentados, a abordagem baseada em aprendizagem de máquina teve performance superior em questão de acurácia.

Algumas outras análises foram realizadas. Entre elas a de que focar em um domínio específico pode resultar em melhores desempenhos do que um classificador genérico.

O trabalho de Noori (2021) consiste em uma análise de sentimentos sobre resenhas de consumidores obtidas a partir do website tripadvisor, um dos maiores e mais utilizados da área de viagens. Um corpus contendo cerca de 400 documentos foi formado a partir desses documentos de texto.

Foi realizado um pré-processamento de dados utilizando a sequência a seguir: Limpeza dos dados, Tokenização, Remoção de StopWords, Stemming, Avaliação do peso de cada termo (Term Weighting), Pruning of the Words e Feature selection.

O coeficiente de Gini foi utilizado para aplicar a Feature Selection, gerando dois subconjuntos de dados, contendo 25 e 100 dos atributos considerados como mais importantes. O original possuía 1892. Os seguintes algoritmos de aprendizagem de máquina foram utilizados na classificação dos textos sobre os três conjuntos: SVM, Redes Neurais, NB, DT, C4.5 and kNN.

A avaliação de performance se deu a partir das seguintes métricas: Cobertura, Precisão, Acurácia, F-Measure e Área da Curva Roc. Os algoritmos DT e C4.5 tiveram os melhores resultados considerando o subconjunto com maior número de features utilizados na etapa de feature selection.

Já em (KRISHNA et al., 2019) é realizada uma análise de sentimentos sobre um conjunto de dados textuais contendo reclamações de consumidores indianos relacionadas à serviços bancários. Foram adquiridas cerca de 2000 reclamações provenientes de 4 bancos distintos. Essas reclamações foram classificadas por especialistas no domínio em moderadas ou graves.

Foram realizadas as seguintes tarefas clássicas de pré-processamento textual: tokenização, remoção de pontuação, remoção de stopwords e stemmização. Além dessas, também foram utilizadas e avaliadas separadamente as seguintes técnicas de feature extraction: TF-IDF, Word2Vec e LIWC.

Os dados foram separados em conjuntos de treinamento e teste, na proporção 80/20. Foi utilizado 10-fold cross validation apenas sobre os 80% do conjunto de treinamento.

Os seguintes algoritmos de aprendizagem de máquina supervisionada foram testados para cada uma das técnicas de extração de feature extraction mencionadas anteriormente: Regressão Logística, Naive Bayes, SVM, Árvore de Decisão, KNN, Random Forest, XGBoost e MLP. A principal métrica utilizada para avaliação dos métodos foi a área sob a curva ROC. Dentre os cenários testados, a técnica LIWC juntamente com os algoritmos de aprendizagem de máquina Random Forest e Naive Bayes tiveram os melhores resultados de acordo com os testes estatísticos realizados.

3.4 CONSIDERAÇÕES FINAIS

De acordo com os trabalhos apresentados nas seções anteriores, elaboramos uma tabela comparativa entre alguns dos atributos considerados relevantes para podermos analisar o que os trabalhos propõem, como são concretizados e a forma com que os mesmos são avaliados.

- **Extração de dados** indica se os trabalhos mencionados detalham como é realizado o processo de extração de informação dos dados que serão utilizados.
- **Natureza dos dados** envolvidos indica se os dados são numéricos/discretos ou somente textuais.
- **Tipo da Transformação do Texto** indica se o trabalho em questão faz algum tipo de transformação para possibilitar que o texto sirva como entrada para os algoritmos de aprendizagem de máquina utilizados.
- **Aprendizagem supervisionada** traz um valor booleano que indica se o trabalho em questão usa alguma técnica com essa abordagem.
- ***Cross validation*** indica se o trabalho utiliza ou não essa técnica.
- ***Deep learning*** indica se o trabalho utiliza ou não essa técnica.
- **Ranqueamento** informa se existe algum tipo de ranqueamento sendo aplicado no trabalho.

- **Tarefas principais** destaca as tarefas que estão sendo executadas em cada um dos trabalhos relacionados.

Tabela 1 - Comparativo entre os Trabalhos

	Rabbi et al	**Ghazzawi, Alharbi**	**Vermeer et al**	**Noori**	**Krishna et al**
Extração de Dados	Não	Não	Não Mencionado	Não	Não
Natureza dos Dados	Numérico / Discreto	Numérico / Discreto	Textual	Textual	Textual
Tipo da Transformação do Texto	Não	Não	TF-IDF	TF-IDF	TF-IDF, Word2Vec, LIWC
Aprendizagem Supervisionada	Sim	Sim	Sim	Sim	Sim
Cross Validation	Não Mencionado	Não Mencionado	Não Mencionado	Sim, 10-Fold	Sim, 10-Fold
Deep Learning	Não	Não	Não	Não	Sim
Ranqueamento	Não	Não	Não	Não	Não
Tarefas Principais	KDD, Clusterização	KDD, Classificação	Classificação, Análise de Sentimento	Classificação, Análise de Sentimento	Classificação, Análise de Sentimento

Fonte: Autor

O capítulo 4, a seguir, irá detalhar o problema abordado bem como a solução proposta neste trabalho. Na seção 4.8, a Tabela 8 repete a tabela 1 acima, incluindo o nosso trabalho, estabelecendo assim uma comparação mais clara entre esses trabalhos e o nosso.

4 PROCESSO DE CONSTRUÇÃO DE CLASSIFICADORES DE RECLAMAÇÕES DE USUÁRIOS

Este capítulo tem por objetivo apresentar detalhadamente o trabalho realizado. Ele é composto por 7 seções. A seção 4.1 explora os problemas que serviram de inspiração e deram origem a esse trabalho. Na seção 4.2, apresentamos uma visão geral do processo proposto neste trabalho. Já na seção 4.3, temos o processo realizado para a extração de informação do “Reclame Aqui”, que deu origem ao do corpus de texto utilizado. A seção 4.4 detalha as técnicas utilizadas para realizar o pré-processamento dos textos.

A seção 4.5 é de importância central, pois apresenta os detalhes relacionados a criação dos modelos de inteligência artificial utilizados na tarefa de classificação, bem como as técnicas utilizadas para a tarefa de ranqueamento. Na seção 4.6 trazemos os testes realizados, seguidos da apresentação dos resultados do ranqueamento (seção 4.7). Por fim, a seção 4.8 traz as considerações finais sobre o processo e o protótipo desenvolvido.

4.1 DETALHAMENTO DO PROBLEMA

As principais questões a serem abordadas neste trabalho podem ser resumidas como:

1. Quais estratégias seriam mais adequadas para minimizar a perda de clientes que efetuaram o registro de reclamações na web?
2. É possível identificar automaticamente as reclamações mais relevantes considerando os dados disponíveis?
3. Qual algoritmo de aprendizagem de máquina é capaz de classificar as reclamações textuais como prioritárias com a melhor performance de acordo com as métricas de precisão, cobertura, acurácia, f-measure e curva roc?
4. Um modelo com menos dados, porém de um ramo de negócio específico traz resultados melhores que um contendo reclamações de empresas com naturezas diversas?

Conforme os trabalhos mencionados na seção 3.1, um dos principais desafios enfrentados nos setores que tratam do atendimento ao consumidor é lidar com a insatisfação de seus clientes. Uma vez que um determinado consumidor possui esse sentimento de insatisfação, ele tem basicamente uma escolha principal a ser realizada: agir ou não sobre esse sentimento.

Caso o consumidor não tome nenhuma atitude para expressar seu descontentamento com uma determinada empresa, a menos que exista algum tipo de comunicação ativa da empresa para com seus clientes, não há nenhuma forma para a empresa tomar conhecimento e posteriormente entender e buscar solucionar o problema apresentado.

Já quando o cliente decide agir, tomando uma atitude sobre sua insatisfação, abre-se um leque de possibilidades; Conforme apresentado na figura 10 da seção 3.1. das diversas atitudes a serem tomadas pelo consumidor, a que representa um menor dano para a empresa é justamente a abertura de reclamação. Uma vez que caso o problema apresentado na reclamação seja resolvido rapidamente e de forma satisfatória, é comum que o relacionamento entre o cliente insatisfeito e a empresa ainda possa ser reconciliado (BLODGETT; WAKEFIELD; BARNES, 1995). Além de que, mesmo nos casos em que esse relacionamento não seja de todo reconciliado, pois o problema existente não foi totalmente resolvido, caso esse mesmo cliente tenha sido respondido de forma rápida e cordial é mais provável que essa insatisfação não escale para consequências mais sérias. Dentre as principais consequências que pretendemos evitar, temos o boca-a-boca negativo, que em sua forma eletrônica tende a se espalhar muito mais rápido e atingir ainda mais pessoas que em sua forma tradicional. Também é muito importante tentar evitar principalmente que essa reclamação chegue a envolver órgãos governamentais de controle ou que tome proporções jurídicas.

A primeira questão de pesquisa deste trabalho consiste em verificar quais estratégias seriam mais adequadas para minimizar a perda de clientes que efetuaram o registro de reclamações na web. Para respondê-la, cogitamos diversas possíveis estratégias que podem vir a ser adotadas para lidar com esse problema, como por exemplo: Responder todas as reclamações, aumentar a agilidade no

processo de aquisição de uma resposta, oferecer alguma vantagem como um crédito ou desconto específico entre outros.

É importante mencionar também que as reclamações coletadas pelas empresas podem ser provenientes de diversas origens distintas, sejam elas sites específicos voltados a reclamações de consumidores, blogs, as diversas redes sociais, além dos canais oficiais da empresa que também oferecem alternativas para o atendimento ao consumidor. Por isso é importante ter em mente que apesar da ideia de simplesmente tentar resolver todas as reclamações o mais rapidamente possível ser interessante, também devemos considerar os diversos canais de comunicação digitais existentes atualmente, pois acabamos tendo que lidar com uma quantidade enorme de reclamações a serem processadas, tornando basicamente inviável tratar em tempo hábil todas as reclamações existentes, seja por limitações de pessoal, recursos financeiros entre outros fatores (RYNGELBLUM et al., 2013). Os custos de uma operação, seja de origem própria ou até mesmo terceirizada, que seja capaz de lidar com tantas reclamações seriam bastante elevados, uma vez que seria necessário uma grande quantidade de pessoas qualificadas e habilitadas a responderem os questionamentos, gerando um alto custo com pessoal da empresa ou com terceirização, sem mencionar gastos com treinamentos/qualificação etc.

4.2 VISÃO GERAL DO PROCESSO PROPOSTO

Nesse contexto, propomos uma abordagem voltada à priorização das reclamações consideradas mais importantes a serem resolvidas, com objetivo de responder as reclamações com maior potencial de escalarem para outras ações mais danosas à empresa como a perda do cliente ou o ingresso de ações em órgãos governamentais de controle e judiciais.

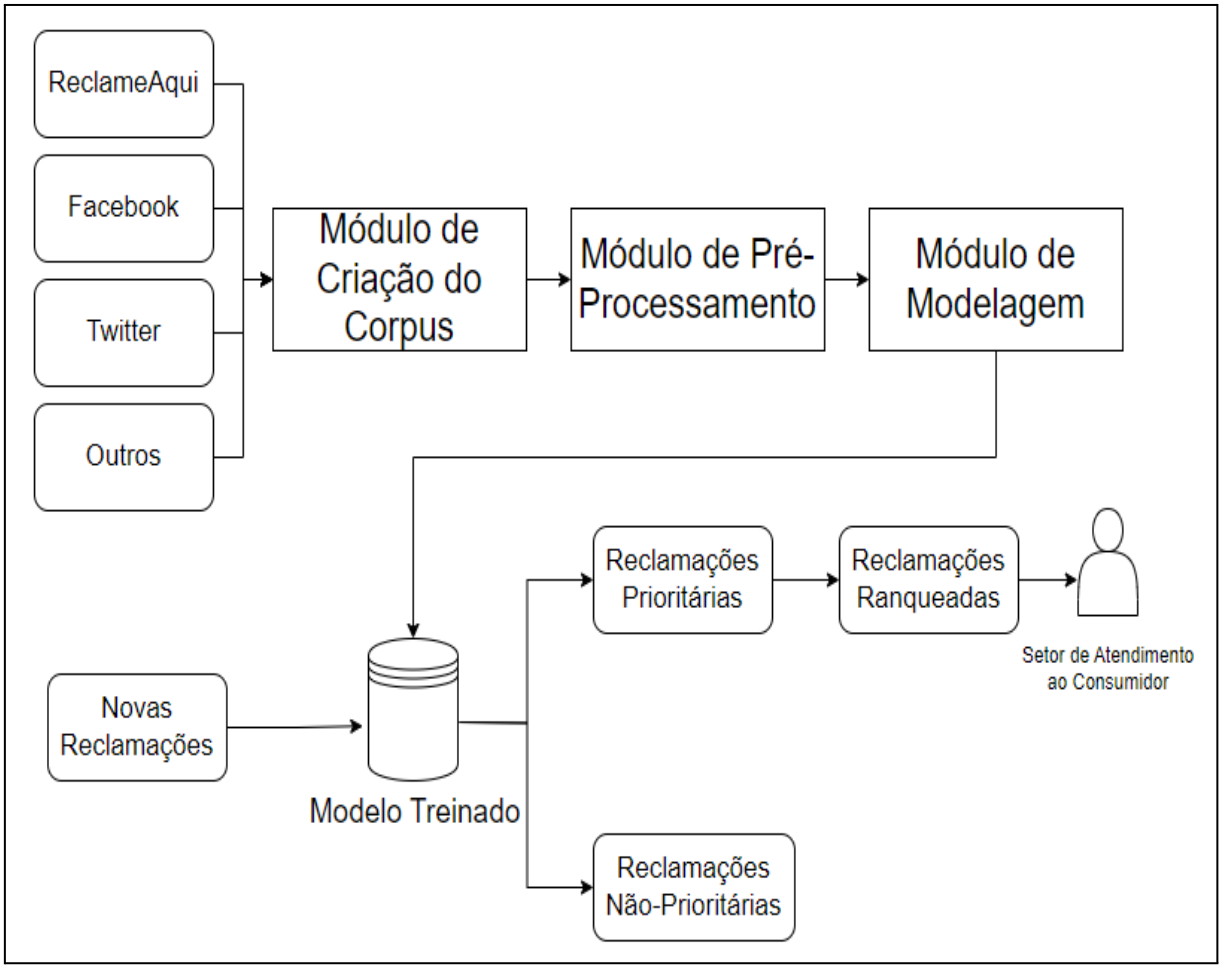
Para tanto, analisamos pesquisas atuais da área e chegamos a um processo capaz de produzir aplicações que tratem adequadamente a primeira questão apresentada, bem como também de realizar experimentos para responder o segundo questionamento.

O processo em questão é composto de diversas etapas que serão detalhadas nas seções posteriores. O trabalho culminou na criação de um protótipo que serve

como exemplo da aplicação deste processo para a resolução prática desse tipo de problema, além de responder também questões de pesquisa 2, 3 e 4, que são mais voltadas à avaliação da qualidade da solução e suas particularidades.

A figura 12, a seguir, retrata uma visão geral dos módulos e do fluxo envolvido no processo proposto.

Figura 12 - Fluxo do processo completo



Fonte: Autor

Conforme podemos observar na figura 12, o fluxo do processo definido neste trabalho funciona da seguinte forma: inicialmente, é necessário treinar o nosso classificador. Para tanto, por trabalharmos com algoritmos de aprendizagem de máquina que realizam a classificação através da abordagem supervisionado, necessitamos de dados textuais etiquetados que alimentem o nosso modelo a ser treinado. Para conseguir obter esses textos de reclamações, foi necessário construir

um extrator de informação (*scrapper*) capaz de acessar essas fontes de reclamações textuais e capturá-las, armazenando os textos de nosso interesse dentro do nosso primeiro módulo. Posteriormente, ainda no módulo de criação de corpus, disponibilizamos as tuplas de informação extraídas: título, link e texto da reclamação para que possam ser devidamente etiquetadas por pessoas especialistas no domínio de atendimento ao consumidor que possam definir de acordo com os interesses de cada a empresa as reclamações que serão consideradas como prioritárias.

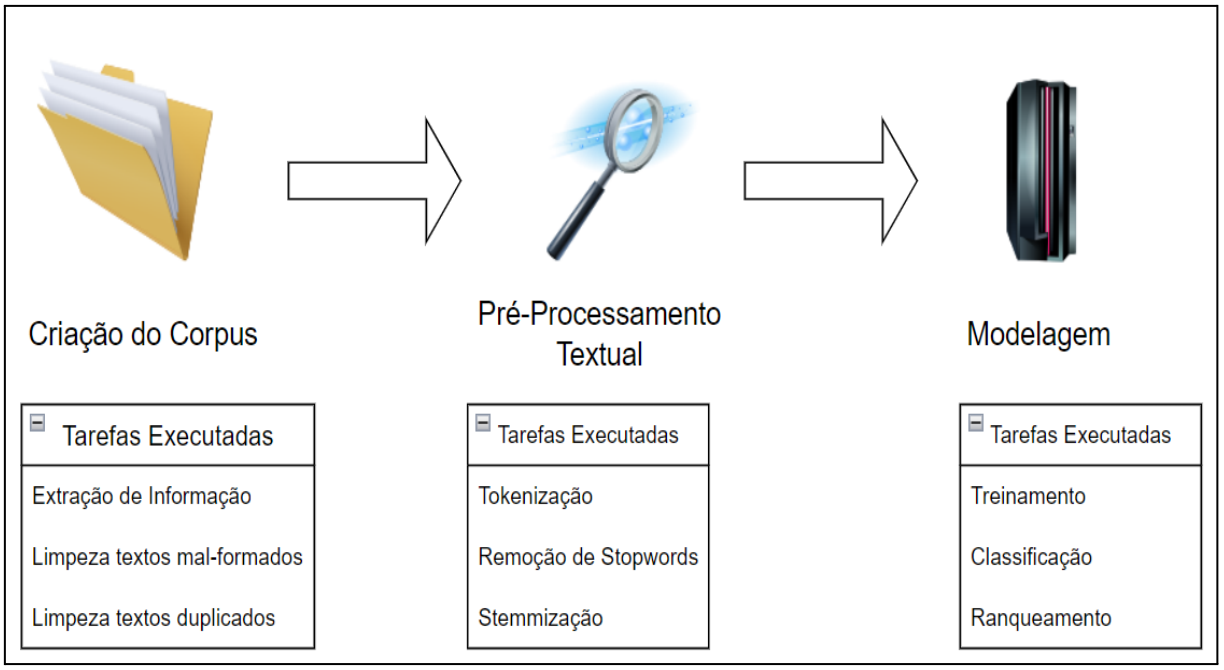
No próximo módulo mencionado no processo, é realizada a etapa de pré-processamento que é responsável por tratar os textos brutos das reclamações, transformando-os em tokens que posteriormente sofrerão um processo de transformação textual para que estejam prontos para alimentar o próximo módulo.

Já o módulo seguinte é responsável pela criação do modelo de aprendizagem de máquina que receberá os dados de treinamento obtidos através dos módulos anteriores e que realizará a concepção de um modelo treinado que posteriormente servirá para efetivamente classificar as novas reclamações recebidas em prioritárias e não prioritárias..

Dessa forma, um atendente do setor de serviço de atendimento ao consumidor ao receber um novo lote de reclamações, não precisará lidar imediatamente com todas elas, e sim atender inicialmente as reclamações que foram classificadas pelo nosso modelo como prioritárias, e de forma sequencial, uma vez que as mesmas foram ranqueadas de acordo com a sua propensão à classe alvo ou seja, de acordo com quão provável seria aquela determinada reclamação ser do tipo prioritária e que portanto devem ser respondidas o mais rápido possível por representar consumidores que expressam um maior grau de insatisfação a partir de seus textos de reclamações.

Na figura 13 a seguir, apresentamos um resumo dos três módulos que compõem o protótipo, a sua sequência de fluxo de dados e as principais tarefas executadas dentro de cada um dos módulos. Nas próximas seções haverá um detalhamento sobre cada uma dessas tarefas mencionadas.

Figura 13 - Visão geral dos módulos do processo



Fonte: Autor

4.3 CRIAÇÃO DO CORPUS

Nesta etapa do processo, executaremos a tarefa de criação do corpus textual que servirá como a entrada para as próximas etapas do processo proposto. É importante mencionar que as técnicas adotadas durante essa etapa poderão variar bastante de acordo com a origem dos documentos desejados à comporem o corpus prestes a ser criado.

Para a criação do corpus a ser adotado no protótipo desenvolvido a partir da aplicação de nosso processo, foram obtidos dados textuais através de reclamações registradas apenas no site Reclame Aqui. No entanto, a utilização de outras fontes de dados conforme sugerido na figura 12, em nada impactaria no andamento das outras etapas descritas no processo. Foi desenvolvido um módulo utilizando a linguagem de programação Python e o framework Selenium para automatizar a tarefa de captura dos textos e extração das informações de interesse.

A utilização do Selenium foi fundamental para o desenvolvimento da tarefa, uma vez que o site do Reclame Aqui é renderizado quase que totalmente via javascript, isto é, de forma dinâmica. Por isso utilizamos o Selenium para simular um

browser utilizando o WebDriver do Google Chrome, possibilitando que todas as informações da página sejam carregadas e portanto tornando-as viáveis de serem extraídas.

Figura 14 - Exemplo de página do Reclame Aqui contendo lista de reclamações

Fonte: Autor

Para realizar a extração das informações desejadas, fazemos análises no código gerado pelo website para encontrar tags de html e seletores css característicos à atributos das reclamações como títulos, links e textos completos. Durante a realização dessa tarefa, acabamos percebendo que os textos completos das reclamações não ficavam disponíveis diretamente na página principal de uma determinada empresa conforme podemos ver um exemplo na figura 14, apenas uma parte do conteúdo textual da reclamação estava sendo renderizado, tornando necessário a travessia pelos links individuais de cada uma das reclamações disponíveis nas páginas principais para conseguir ter acesso à todo o conteúdo textual presente nas mesmas. Esse processo de realizar a visita de link por link em cada uma das reclamações registradas, cujo conteúdo completo desejamos extrair, também foi automatizado no módulo desenvolvido através do Selenium.

Figura 15 - Exemplo de parte do código renderizado pelo site do Reclame Aqui

Fonte: Autor

Dessa forma, para realizar a captura e extração das informações textuais disponíveis no módulo desenvolvido, é necessário apenas informar como parâmetros a empresa que será objeto da captura dos dados, juntamente com a quantidade de páginas sobre as quais iremos realizar o processo de extração das informações textuais. Como resultado teremos listas contendo os títulos das reclamações, os seus respectivos links e o texto complementar com o conteúdo da reclamação em si.

Figura 16 - Exemplo de listas resultantes do processo de extração de informação.

Title	Links	Text
Meu Nike perdeu o ar air	.../netshoes/meu-ni...	Meu tênis com pouco tempo de uso perdeu o ar d...
Preciso tirar meu CPF da conta do meu irmao	... /netshoes/precis...	Boa noite eu fis um cadastro na netshoes para ...
Demora na troca de produto	...netshoes/demora...	Solicitei a troca de 1 sapato no dia 11/05. Po...
Fiz a devolução e não recebi o estorno do valor	... /netshoes/fiz-a-...	Fiz uma compra no dia 18/04, como o tênis não ...
Devolução do tênis mizuno na loja Kanguroos	.../netshoes/devolu...	Eu fiz uma compra na Netshoes, juntamente com ...

Fonte: Autor

As listas resultantes ainda podem ser facilmente armazenadas em um arquivo do tipo csv, ou seja, que pode ser lido em um formato de planilha. Esse tipo de

arquivo gerado funciona como uma base de dados extremamente simples e facilmente manipulável para persistência de todos os dados obtidos no módulo.

Antes de realizar o salvamento dos textos obtidos, são realizadas duas simples checagens. A primeira tem como objetivo verificar se ocorreu algum erro no processo de extração, resultando em algum dos atributos de interesse não sendo preenchidos. E a segunda verificando se houve algum tipo de duplicidade nos dados extraídos. As reclamações que apresentarem algum dos problemas mencionados serão consideradas ruidosas, e portanto, descartadas.

Uma vez que já temos os dados textuais de interesse disponíveis e armazenados em nossa base de dados, passamos para a última etapa da criação do corpus de dados utilizado no protótipo, a de anotação ou etiquetamento dos textos obtidos com as classes de interesse.

O processo de anotação dos textos extraídos foi realizado manualmente por pessoas com experiência no domínio de atendimento ao consumidor. Cada tupla contendo título, link e texto foram rotulados com apenas um entre os dois valores possíveis de serem assumidos: sendo considerados prioritários ou não-prioritários.

Foram extraídas um total de 1496 reclamações na forma de tuplas textuais, das quais após realizar o processo de limpeza mencionado anteriormente restaram 1489 tuplas que foram devidamente rotuladas. As reclamações possuem uma média e mediana de 611,15 e 458 caracteres respectivamente. No quadro abaixo apresentamos as 27 empresas de 6 segmentos distintos, das quais as reclamações extraídas são originárias.

Quadro 4 - Empresas que possuem reclamações no corpus

Telefonia	**Tecnologia**	**Bancos / Financeiras**
Tim	Kabum	Bradesco
Claro	Samsung	Itaú
Vivo	Apple	Santander
Oi	HP	Nubank
		Banco do Brasil

Departamento	Redes Sociais	Vestuário / Calçados
Extra	Facebook	Shein
Americanas	Instagram	Netshoes
Kalunga	Tiktok	
Submarino	Twitter	
Magazine Luiza	Kwai	
Casas Bahia		
Amazon		

Fonte: Autor

O corpus resultante através da aplicação do processo em nosso protótipo está disponível para acesso público através de uma página disponibilizada pelo autor[1].

4.4 PRÉ-PROCESSAMENTO

Foi desenvolvido um módulo específico para a realizar o pré-processamento dos textos obtidos através do módulo de captura dos dados e criação do corpus conforme mencionado na seção anterior.

No módulo de pré-processamento são executadas diversas tarefas necessárias para transformar os textos capturados em dados passíveis de serem utilizados por algoritmos de classificação baseados em aprendizagem de máquina.

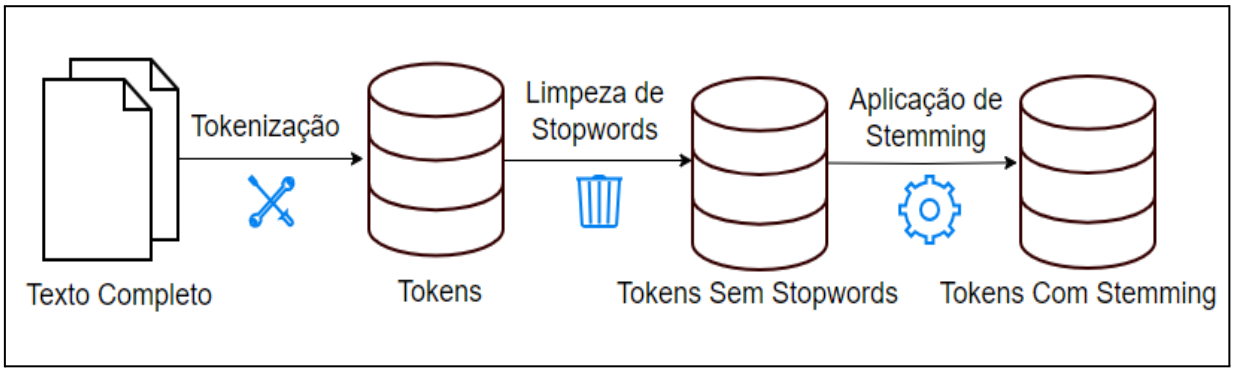
A implementação foi realizada de forma subdividida em duas etapas distintas. Na primeira etapa foram realizadas as seguintes tarefas: Tokenização, Remoção de StopWords e Stemmização. Os tokens gerados como subproduto da execução dessa etapa também são salvos no arquivo .csv para persistência e utilização nas etapas posteriores deste processo.

Na figura 17 apresentamos a sequência das tarefas executadas durante o pré-processamento, bem como os subprodutos gerados depois da execução de cada uma das tarefas.

1

[1] https://github.com/gabrielh10/complaints_reclameaqui

Figura 17 - Tarefas do Pré-Processamento



Fonte: Autor

É válido mencionar que durante a execução da tarefa de remoção de stopwords, também realizamos juntamente a remoção de tokens contendo caracteres considerados como indesejáveis, como por exemplo, tokens contendo espaços vazios e sinais de pontuação.

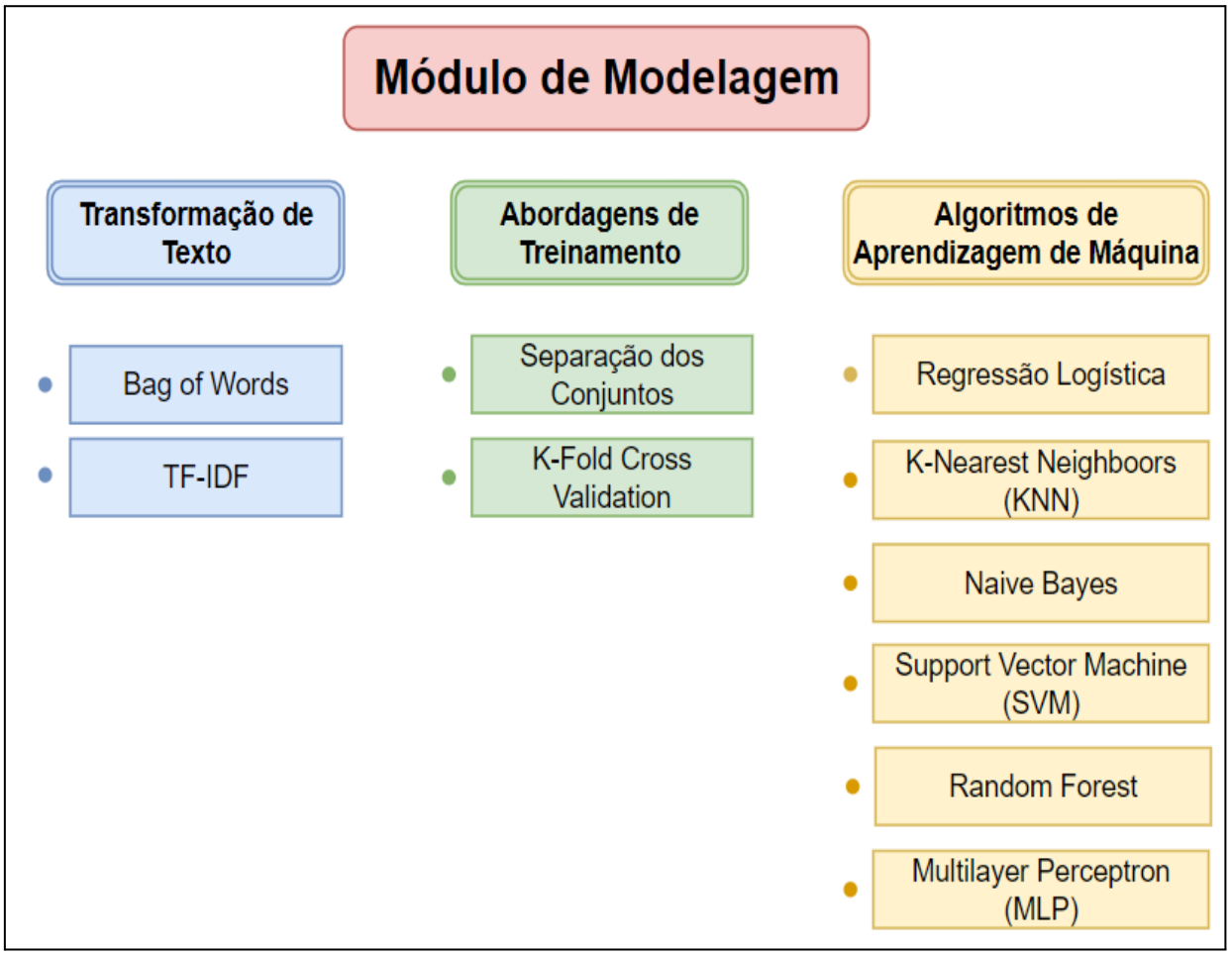
4.5 MODELAGEM

Para implementação da etapa de modelagem, o primeiro fator a ser considerado foi definir a estratégia que será utilizada para a transformação de texto. No nosso protótipo permitimos a escolha de uma entre as seguintes abordagens: Bag-of-Words e Transformação TF-IDF.

Este módulo utiliza os tokens já devidamente processados na etapa anterior e convertidos através de uma das técnicas de transformação de texto disponíveis para alimentar os algoritmos de aprendizagem de máquina supervisionados que serão utilizados para realizar a classificação das reclamações.

Todo esse processo foi implementado num módulo específico para treinamento e classificação também utilizando a linguagem de programação Python, mas contando com a utilização da biblioteca Scikit-Learn. É uma biblioteca open source que possui diversos algoritmos de aprendizagem de máquina já implementados e prontos para serem utilizados com diversas possibilidades de parametrização.

Figura 18 - Opções de Modelagem



Fonte: Autor

Apesar da transformação de texto não ser exatamente uma etapa considerada como parte do processo de modelagem, optamos por incluí-la nesse módulo para manter a independência entre os módulos. Na verdade, trata-se de uma tarefa “intermediária”, entre a saída do módulo de pré-processamento, que são tokens processados, e a entrada dos algoritmos de aprendizagem de máquina, que são numéricas e consistem numa forma representação dos tokens previamente mencionados.

4.5.1 Treinamento

Para o treinamento utilizamos os textos já processados e que sofreram uma transformação textual seja pela abordagem BoW ou TF-IDF como entrada para os

algoritmos de aprendizagem de máquina supervisionada que iremos utilizar na classificação.

Testamos duas abordagens distintas para realizar o treinamento:

1. Separação dos conjuntos em treinamento e teste: Que consiste em separar todos os textos processados em dois conjuntos distintos seguindo as seguintes proporções: 70% para o treinamento e 30% para o conjunto de teste.
2. K-Fold Cross Validation: Consiste em particionar o conjunto em “K” folds, sendo K-1 utilizados para treinamento e o restante para teste. Porém esse processo é repetido “K” vezes, alterando a cada execução o fold utilizado para teste. Na prática, o parâmetro “K” utilizado no protótipo foi igual a 10.

4.5.2 Classificação

Nesta etapa executamos os experimentos utilizando os algoritmos de aprendizagem de máquina supervisionados mencionados na seção 2.3.1 e que passaram pelo processo de treinamento descrito na seção anterior.

Os algoritmos foram executados com os seguintes parâmetros:

- Regressão Logística: parâmetros padrões (C=1; Tol=0,0001; Penalty = “L2”).
- KNN: A métrica utilizada para cálculo da distância pelo KNN foi a distância de Minkowski que é basicamente uma generalização entre as distâncias Euclidiana e de Manhattan. O outro parâmetro importante a ser definido para a execução do algoritmo é a quantidade de vizinhos a serem considerados. Na implementação de nosso protótipo o valor 3 foi selecionado para o parâmetro.
- Naive Bayes: parâmetros padrões (alpha = 1.0).
- SVM: A função de kernel utilizada foi a RBF (Radial Basis Function).
- Random Forest: O coeficiente de Gini foi utilizado como critério de separação
- MLP (Deep Learning): Ao todo temos uma arquitetura com 6 camadas. Sendo uma delas a camada de entrada. Além desta, foram utilizadas 4 camadas

escondidas com respectivamente 256, 128, 64 e 32 neurônios em cada uma dessas camadas. Por fim, temos a última camada, a de saída..

4.5.3 Ranqueamento

Esta é a última tarefa a ser realizada dentro do módulo responsável pela etapa de modelagem, nela realizamos o ranqueamentos dos dados classificados como pertencentes à classe alvo (textos considerados como prioritários). O objetivo dessa etapa é gerar uma lista contendo as reclamações que serão atendidas respeitando a ordenação realizada.

Para tanto, realizamos a atribuição de um score para cada um dos textos e realizamos uma ordenação decrescente a partir deste mesmo atributo, de forma que as reclamações de scores mais altos, que representam justamente as reclamações de consumidores com maior grau de propensão a classe alvo (i.e., com maior probabilidade de serem realmente prioritárias), sejam atendidas mais rapidamente.

A atribuição do score que servirá como base para o ranqueamento foi realizada tendo em vista a seguinte abordagem: utilizamos uma regressão linear sobre os textos das reclamações de forma a gerar o score de propensão à classe alvo, que será o atributo utilizado para a ordenação dos textos. Quanto maior esse valor, ou seja, mais próximo de 1, maior a probabilidade de pertencer à classe das reclamações prioritárias, e portanto, sendo considerado como de maior relevância.

Por fim, nesta etapa, oferecemos a possibilidade de exportar os resultados, ou seja, a lista de reclamações prioritárias devidamente ranqueadas nesta etapa para uma planilha que pode ser utilizada diretamente pelos funcionários responsáveis pelas respostas aos consumidores ou ainda servirem para alimentar sistemas próprios das empresas.

4.6 TESTES REALIZADOS

Nesta seção apresentamos os resultados dos diversos experimentos realizados neste trabalho. Também expomos comentários pertinentes sobre as informações apresentadas.

É necessário informar que o conjunto de dados usado nos testes realizados é na verdade um subconjunto balanceado do corpus criado contendo 720 textos de reclamações, ou seja, temos 360 elementos da classe prioritária que representa as reclamações consideradas prioritárias e 360 da classe complementar que é composta pelas reclamações não-prioritárias. .

Conforme mencionado na seção anterior, realizamos testes utilizando duas abordagens distintas para definição do conjunto de treinamento.

4.6.1 Testes com separação dos conjuntos em treinamento e teste

Na tabela 2 apresentamos os resultados obtidos na classificação dos textos prioritários utilizando para tanto Bag-of-Words para transformação de texto e a abordagem de separação dos conjuntos em treinamento e teste de acordo com as métricas citadas neste trabalho.

Tabela 2 - Resultados da classificação com separação dos conjuntos e BoW

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,7282	0,7895	0,7787	0,7576	0.8474
KNN	0,1068	1	0.5741	0,1930	0.5827
Naive Bayes	**0,8641**	0,7008	0.7593	0,7739	0.8444
SVM	0,6990	0,7579	0.7500	0,7273	0.8374
Random Forest	0,7670	0,7900	**0,7917**	**0,7783**	**0,8560**
MLP (Deep Learning)	0,6796	**0,8046**	0,7685	0,7368	0,8168

Fonte: Autor

Já na tabela 3 temos os resultados da execução dos mesmos algoritmos de aprendizagem de máquina sobre o mesmo conjunto de textos, porém dessa vez, utilizamos TF-IDF para transformação de texto e mantivemos a abordagem de separação dos conjuntos no treinamento dos modelos.

Tabela 3 - Resultados da classificação com separação dos conjuntos e TF-IDF

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,8447	0,7436	0,7870	0,7909	0.8563
KNN	0,7282	0,7009	0.7222	0,7143	0.7594
Naive Bayes	**0,8835**	0,6791	0.7454	0,7679	**0.8587**
SVM	0,8350	0,7414	0.7824	0,7854	0.8539
Random Forest	0,7864	**0,8020**	**0,8056**	**0,7941**	0,8556
MLP (Deep Learning)	**0,8835**	0,7165	0,7778	0,7913	0,8583

Fonte: Autor

Conforme podemos inferir analisando os resultados apresentados, a abordagem de transformação de texto TF-IDF mostrou resultados superiores para quase todos os algoritmos de aprendizagem de máquina treinados a partir da abordagem de treinamento baseada na separação dos conjuntos em treinamento e teste em basicamente todas as nossas métricas de interesse.

4.6.2 Testes com K-Fold Cross Validation

Com essa informação em mente, optamos por realizar o mesmo tipo de teste, porém dessa vez utilizando a abordagem de K-Fold Cross Validation no treinamento dos modelos em vez da separação dos conjuntos em treinamento e teste. Ou seja, a realização de um novo experimento contando com os mesmos algoritmos de aprendizagem de máquina utilizados no experimento anterior, porém isolando as técnicas de transformação de texto BoW e TF-IDF, para verificar qual delas é capaz de produzir resultados superiores de acordo com nossas métricas. Na tabela 4 temos os resultados utilizando a abordagem de transformação de texto BoW.

Tabela 4 - Resultados da classificação com 10-fold cross validation e BoW

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,7606	**0,8375**	**0,8056**	**0,7943**	0.8834
KNN	0,1109	0,8125	0.5514	0,1916	0.6255
Naive Bayes	**0,7946**	0,7743	0.7833	0,7833	0.8358
SVM	0,7448	0,7931	0.7764	0,7670	0.8699
Random Forest	0,7636	0,8222	0,8000	0,7893	**0,8856**
MLP (Deep Learning)	0,7382	0,8320	0,7931	0,7756	0,8528

Fonte: Autor

Já na tabela 5 apresentamos os resultados provenientes da classificação utilizando a abordagem TF-IDF na tarefa de transformação de texto e mantendo 10-fold cross validation como estratégia de treinamento.

Tabela 5 - Resultados da classificação com 10-fold cross validation e TF-IDF

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,8391	0.8122	**0,8222**	**0,8239**	**0,8881**
KNN	0,7045	0,7060	0,7083	0,7037	0,7739
Naive Bayes	**0,8600**	0,7603	0,7931	0,8047	0,8708
SVM	0,8274	**0,8165**	0,8208	0,8208	0,8854
Random Forest	0,7686	0,8086	0,7944	0,7853	0,8764
MLP (Deep Learning)	0,8157	0,7598	0,7764	0,7820	0,8726

Fonte: Autor

De acordo com a análise dos resultados das tabelas 4 e 5, percebe-se que a abordagem TF-IDF no geral também manteve os melhores resultados, porém é importante mencionar que ao utilizarmos 10-fold cross validation no processo de

treinamento, a superioridade da abordagem tf-idf na transformação de texto não aparece de forma notável, uma vez que durante a execução do algoritmo random forest a abordagem BoW acabou tendo melhores resultados. Também é importante mencionar que no algoritmo de deep learning MLP os resultados foram mistos, ou seja, dependendo da métrica de interesse, a abordagem de transformação de texto capaz de produzir os melhores resultados varia.

A regressão logística e o SVM se destacaram positivamente na configuração composta por 10-Fold Cross Validation e e TF-IDF, uma vez que foram capazes de obter resultados acima da casa de 0,80 para todas as métricas disponíveis.

Por outro lado, os piores resultados dos nossos experimentos foram encontrados na execução do algoritmo de aprendizagem de máquina KNN em conjunto com a abordagem de treinamento BoW. A execução do KNN com as abordagens de treinamento de separação de conjuntos e 10-Fold Cross Validation, tiveram coberturas de 10,68% e 11,09% respectivamente. Valores muito abaixo se comparados aos outros algoritmos de aprendizagem de máquina executados nestes experimentos.

4.6.3 Testes com Corpus de domínio específico

Por fim, um outro experimento foi realizado utilizando um corpus menor contendo apenas textos de reclamações realizadas dentro de apenas um domínio específico, no caso, o domínio escolhido foi o de empresas de telecomunicação. Dessa forma, foi formado um subconjunto balanceado de 198 reclamações, isto é, 99 textos representantes para cada uma das classes, composto exclusivamente por reclamações de uma das seguintes empresas ligadas ao setor de telecomunicação: Tim, Claro, Oi e Vivo. Queríamos avaliar se a performance dos algoritmos de aprendizagem seria superior em um domínio específico mesmo que em contrapartida tenha um menor volume de dados disponíveis para a etapa de treinamento. Por conta da quantidade já reduzida de textos disponíveis, optamos por utilizar 10-Fold Cross Validation no treinamento em vez da abordagem de separação dos conjuntos.

Na tabela 6 temos os resultados da execução dos mesmos algoritmos de aprendizagem de máquina utilizados nos experimentos anteriores, utilizando a abordagem BoW sobre o subconjunto textual denominado telecomunicações.

Tabela 6 - Resultado da classificação do subconjunto com 10-Fold Cross Validation e BoW

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,5717	0,7131	0,6771	0,6274	0,7486
KNN	0,0724	0,5000	0,5355	0,1236	0,6651
Naive Bayes	**0,7212**	0,6466	0.6687	**0,6747**	0,7166
SVM	0,6017	0,6971	0,6821	0,6395	0,7571
Random Forest	0,5709	**0,7693**	**0,6974**	0,6486	**0,7913**
MLP (Deep Learning)	0,3569	0,7700	0,6168	0,4699	0,5904

Fonte: Autor

Já na tabela 7 temos os resultados do experimento utilizando dessa vez a abordagem TF-IDF sobre o subconjunto de telecomunicações na transformação textual.

Tabela 7 - Resultado da classificação do subconjunto com 10-Fold Cross Validation e TF-IDF

Algoritmo de Aprendizagem	Cobertura	Precisão	Acurácia	F-Measure	AUC
Regressão Logística	0,6606	0,6952	0,6679	0,6569	0,7624
KNN	0,7029	0,5937	0,6076	0,6363	0,6493
Naive Bayes	**0,8461**	0,6006	0,6224	**0,6840**	0,7456
SVM	0,6092	0,7258	0,6624	0,6342	0,7657
Random Forest	0,5800	**0,8187**	**0,7226**	0,6702	**0,7846**
MLP (Deep Learning)	0,6032	0,6803	0,6579	0,6298	0,6949

Fonte: Autor

De acordo com os resultados demonstrados nas tabelas 6 e 7, ficou claro que o baixo desempenho encontrado nos algoritmos de aprendizagem de máquina testados sobre o subconjunto das reclamações do setor de telecomunicações, se comparado com os resultados obtidos sobre o conjunto contendo reclamações de diversos setores. A diferença entre o algoritmo de aprendizagem de máquina com melhor desempenho no conjunto de reclamações completo em relação ao conjunto apenas com as reclamações do setor de telecomunicações pode chegar até 15% na métrica F-measure, 10 pontos na área sob a curva ROC, além de quase 10% na métrica de acurácia.

Entre os algoritmos testados com a abordagem BoW, cujos resultados estão presentes na tabela 6, o Naive Bayes e o Random Forest tiveram os melhores resultados na maior parte das métricas de interesse.

A mesma tendência se repetiu nos algoritmos que utilizaram a abordagem TF-IDF na transformação de texto, cujos resultados foram apresentados na tabela 7. Naive Bayes e Random Forest se revezavam no topo dos melhores resultados em todas as métricas disponíveis para o processo de avaliação.

Na comparação direta entre as abordagens de transformação de texto BoW e TF-IDF sobre o subconjunto de reclamações do setor de telecomunicações, percebeu-se uma vantagem considerável dos algoritmos de aprendizagem de máquina que usaram abordagem TF-IDF. A diferença se torna ainda mais notória ao avaliarmos o algoritmo KNN, cujos resultados da classificação na abordagem BoW ficaram mais de 50 pontos percentuais atrás da abordagem TF-IDF ao analisarmos a métrica F-Measure por exemplo.

4.7 APRESENTAÇÃO DOS RESULTADOS RANQUEADOS

Conforme foi mostrado na figura 12, a saída do protótipo desenvolvido consiste nos textos das reclamações considerados como prioritários pelo classificador, juntamente com seus títulos e links de origem respectivos. A ordem de apresentação dos textos é dada a partir do resultado do ranqueamento realizado.

Os textos prioritários, ordenados de acordo com a prioridade, também podem ser facilmente exportados para um formato ”.csv”. Através do qual podem ser utilizados de forma imediata como uma planilha, para que os textos das reclamações sejam analisados e respondidos diretamente através do link de origem da reclamação por um ser humano responsável (ver Figura 19).

Figura 19 - Resultados do ranqueamento

Ranking	Title	Links	Text
1	Indignação pela falta de respeito com o consum...	https://www.reclameaqui.com.br/tim-celular/ind...	No dia 24/08 pedi a portabilidade do meu númer...
2	Portabilidade para Oi	https://www.reclameaqui.com.br/oi-movel-fixo-t...	Olá, recebi uma ligação no dia 24/08/2021 ás 1...
3	Péssimo atendimento	https://www.reclameaqui.com.br/nubank/pessimo-...	Entrei em contato com a empresa pra tirar uma ...
4	PORTABILIDADE (ARREPENDIMENTO)	https://www.reclameaqui.com.br/claro/portabili...	DIA 08 DE MARÇO DE 22 RESOLVI FAZER A PORTABIL...
5	Bolsa é [Editado pelo Reclame Aqui]	https://www.reclameaqui.com.br/itau/bolsa-e_Br...	Tenho conta no Itaú a mais de 13 anos .E sempr...
...	...	...	...
212	anel brilhoso	https://www.reclameaqui.com.br/shein/anel-bril...	comprei um anel de compromisso na shein e deu ...
213	Não fui reembolsada	https://www.reclameaqui.com.br/shein/nao-fui-r...	Fiz a compra no site da Shein no dia 21 de abr...
214	Mercadoria consta q foi entregue não recebi	https://www.reclameaqui.com.br/amazon/mercador...	Eu fiz uma compra e consta q foi entregue e eu...
215	Meu pedido está parado em Curitiba	https://www.reclameaqui.com.br/shein/meu-pedid...	Meu pedido da Shein está parado em Curitiba de...
216	Taxa da receita federal	https://www.reclameaqui.com.br/shein/taxa-da-r...	Fui taxada, paguei a taxa e gostaria de recebe...

Fonte: Autor

. O arquivo “.csv” pode ser usado como uma etapa intermediária, servindo para alimentar um banco de dados ou um outro sistema interno da empresa, que irá alocar as reclamações classificadas como importantes e já ranqueadas para seus colaboradores responderem de acordo com suas regras internas próprias.

4.8 CONSIDERAÇÕES FINAIS

Neste capítulo apresentamos em detalhes durante as seções anteriores todas as etapas que compõem o processo para construção de aplicações capazes de classificar e ranquear as reclamações dos usuários ou clientes de determinada empresa ou serviço. Ao adicionarmos o nosso trabalho ao comparativo de trabalhos relacionados presentes na tabela 1, chegamos a tabela 8.

Tabela 8 - Comparativo entre os trabalhos considerando este

	Rabbi et al	**Ghazzawi, Alharbi**	**Vermeer et al**	**Noori**	**Krishna et al**	**Este Trabalho**
Extração de Dados	Não	Não	Não Mencionado	Não	Não	Sim
Natureza dos Dados	Numérico / Discreto	Numérico / Discreto	Textual	Textual	Textual	Textual
Tipo da Transformação do Texto	Não	Não	TF-IDF	TF-IDF	TF-IDF, Word2Vec, LIWC	BoW, TF-IDF
Aprendizagem Supervisionada	Sim	Sim	Sim	Sim	Sim	Sim
Cross Validation	Não Mencionado	Não Mencionado	Não Mencionado	Sim, 10-Fold	Sim, 10-Fold	Sim, 10-Fold
Deep Learning	Não	Não	Não	Não	Sim	Sim
Ranqueamento	Não	Não	Não	Não	Não	Sim
Tarefas Principais	KDD, Clusterização	KDD, Classificação	Classificação, Análise de Sentimento	Classificação, Análise de Sentimento	Classificação, Análise de Sentimento	Classificação, Ranqueamento

Fonte: Autor

Conforme podemos observar a partir da tabela 8, o grande diferencial deste trabalho consiste na elaboração de um processo completo composto por diversas etapas que realizam a execução de diversas tarefas distintas para a resolução de um tipo de problema específico, o da priorização de reclamações a partir de seus textos.

Além disso, também trazemos os resultados de testes comparativos realizados com diversos algoritmos de aprendizagem de máquina utilizados durante a tarefa de classificação, bem como seus respectivos detalhes, como configurações e parâmetros utilizados durante os experimentos. Testamos também os resultados

dos algoritmos de aprendizagem mediante duas abordagens distintas de treinamento, a de separação dos conjuntos de treinamento e teste na proporção 70-30 e K-Fold Cross Validation utilizando o parâmetro K com valor igual a 10.

De acordo com nossas métricas de interesse, a configuração que se utilizava da transformação de texto por TF-IDF, treinamento utilizando 10 Fold Cross validation e Regressão Logística como algoritmo de aprendizagem tiveram os melhores resultados com 0,8391 de cobertura, 0,8122 de precisão, 0,8222 de acurácia, 0,8239 de f-measure e 0,8881 de área sob a curva roc.

Ao analisarmos os resultados do experimento com um subconjunto de reclamações focadas em um setor específico em comparação com o conjunto de maior volume com reclamações de empresas de setores variados, percebemos que para quantitativos de reclamações próximos aos apresentados neste trabalho, utilizar o conjunto de textos completo tende a resultar num desempenho consideravelmente superior.

Quanto ao ranqueamento realizado entre os textos considerados como prioritários durante a classificação, apresentamos inicialmente apenas uma abordagem relativamente simples para a construção dos rankings: a de regressão linear. Como foi implementada apenas uma estratégia de ranqueamento, não foi realizado nenhum tipo de comparação entre abordagens, o que inclusive será abordado como uma possibilidade na seção de trabalhos futuros.

5 CONCLUSÃO

Neste capítulo sintetizamos as principais contribuições realizadas durante a execução deste trabalho, bem como discutimos uma série de sugestões interessantes para trabalhos futuros.

Inicialmente, partimos de uma questão de pesquisa ampla, que buscava encontrar estratégias que seriam adequadas para minimizar a perda de clientes. A partir das opções consideradas, focamos em verificar se seria possível classificar os textos das reclamações em prioritários e não-prioritários a partir de nosso conjunto de dados. Uma vez que isso mostrou-se possível, evoluímos a questão para definir quais técnicas tinham melhores resultados de acordo com métricas definidas na tarefa de classificação das reclamações. Nesse contexto, percebemos que tínhamos um processo bem definido que envolvia extração, classificação e priorização de textos de reclamações provenientes dos usuários no segmento de atendimento ao consumidor.

5.1 PRINCIPAIS CONTRIBUIÇÕES

Foi desenvolvido um protótipo composto por 3 módulos independentes. O primeiro deles utiliza técnicas de extração de informação para obter os textos de reclamações através do site do ReclameAqui e consolidá-los numa planilha. O segundo módulo é responsável por realizar um pré-processamento textual, transformando os textos completos de reclamações em tokens sem stopwords e stemming aplicado. O último módulo possibilita executar alguns algoritmos de aprendizagem de máquina supervisionados para classificar textos, e uma técnica de ranqueamento para priorização dos textos.

Durante a execução da etapa de extração, geramos um Corpus contendo 1489 textos de reclamações de usuários obtidos através do ReclameAqui. O Corpus em questão possui textos de 27 empresas distintas e que atuam em segmentos diversos como: telefonia, tecnologia, bancos e instituições financeiras, vestuário, lojas de departamento e redes sociais.

Além disso, também fomos capazes de realizar diversos experimentos utilizando o nosso corpus gerado. Entre eles temos, avaliar a performance dos algoritmos de aprendizagem de máquina implementados no nosso protótipo

desenvolvido em relação a duas técnicas de transformação de texto, BoW e TF-IDF. Outro experimento realizado busca avaliar os algoritmos de aprendizagem de máquina utilizados de acordo com duas abordagens distintas de treinamento a serem comparadas: K-Fold Cross Validation e a de separação dos conjuntos. Por fim, realizamos experimentos comparando o desempenho dos algoritmos de aprendizagem de máquina treinados com o corpus completo em relação ao desempenho de um subconjunto formado por textos de reclamações que se encaixam dentro de apenas um domínio específico, no caso, optamos pelo de telecomunicações.

5.2 TRABALHOS FUTUROS

Um possível trabalho futuro seria utilizar outros métodos ou técnicas para realizar o ranqueamento dos textos das reclamações, possibilitando inclusive que se efetuem comparações entre diferentes técnicas de ranqueamento dentro do escopo do trabalho em questão.

Poderíamos também analisar e comparar a performance dos algoritmos de classificação com a utilização de modelos de word embeddings como, por exemplo, Word2Vec. O mesmo também poderia ser feito através da utilização de modelos pré-treinados como o BERT. Seguindo a mesma linha de raciocínio, Few-Shot Learning também seria uma opção interessante, especialmente nos testes realizados sobre corpus de domínio específico, que lidam com uma menor quantidade de dados para treinamento..

Outra possibilidade de trabalho futuro seria aplicar a técnica de ensemble para combinar classificadores e comparar com os resultados obtidos pelos outros algoritmos de aprendizagem de máquina na tarefa de classificação. Testes estatísticos também poderiam vir a ser utilizados na realização das comparações.

Também seria interessante desenvolver uma ferramenta específica que auxilie durante a realização da anotação dos textos que irão formar o corpus final, facilitando principalmente a dificuldade latente quanto a paralelização do trabalho de etiquetar os textos, além de possibilitar também o emprego de uma interface visual mais adequada e amigável para a realização de um processo que normalmente acaba sendo longo e demorado, por ser realizado de forma manual.

REFERÊNCIAS

ABDULLATEEF, A. O., & Salleh, S. M. (2013). **Does customer relationship management influence call centre quality performance? An empirical industry analysis.** Total Quality Management & Business Excellence, 24(9-10), 1035–1045.

AKUMA, S., LUBEM, T. & ADOM, I.T. **Comparing Bag of Words and TF-IDF with different models for hate speech detection from live tweets**. *Int. j. inf. tecnol.* 14, 3629–3635 (2022).

ALLAHYARI, Mehdi et al. **A brief survey of text mining: Classification, clustering and extraction techniques**. arXiv preprint arXiv:1707.02919, 2017.

ALUÍSIO, Sandra Maria; DE BARCELLOS ALMEIDA, Gladis Maria. **O que é e como se constrói um corpus? Lições aprendidas na compilação de vários corpora para pesquisa linguística**. Calidoscópio, v. 4, n. 3, p. 156-178, 2006.

ANANDARAJAN, M., HILL, C., NOLAN, T. (2019). Text Preprocessing. In: Practical Text Analytics. **Advances in Analytics and Data Science**, vol 2. Springer, Cham.

BLODGETT, J. G., WAKEFIELD, K. L., & BARNES, J. H. (1995). *The effects of customer service on consumer complaining behavior.* ***Journal of Services Marketing****, 9(4), 31–42.*

BOWEN, J. T., & CHEN, S. (2001). The relationship between customer loyalty and customer satisfaction. **International Journal of Contemporary Hospitality Management**, 13(5), 213–217.

BREIMAN, L. (2001). Random Forests. **Machine Learning**, 45(1), 5–32.

CHAUVEL, Marie Agnes; GOULART, Vania Cianni. How to use Customer Service Departments to create more value for customers: a review of Brazilian studies/Como gerar valor para os clientes por meio dos servicos de atendimento ao consumidor: o que mostram as pesquisas. **Cadernos EBAPE. BR**, v. 5, n. 4, 2007.

CHEN, Minmin et al. **An alternative text representation to tf-idf and bag-of-words**. arXiv preprint arXiv:1301.6770, 2013.

CHOWDHARY, K. R. **Fundamentals of artificial intelligence**. New Delhi: Springer India, 2020.

CHOWDHURY, G. G. (2005). Natural language processing. **Annual Review of Information Science and Technology**, 37(1), 51–89.

DA CAMARA, N.Z. Identity, Image and Reputation. In: Helm, S., Liehr-Gobbers, K., Storck, C. (eds) **Reputation Management**. Management for Professionals. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-19266-1_6 p. 47−58, 2011.

DE LEANIZ, Patricia Martínez García; DEL BOSQUE RODRÍGUEZ, Ignacio Rodríguez. Corporate image and reputation as drivers of customer loyalty. **Corporate Reputation Review**, v. 19, p. 166-178, 2016.

Dewa Made Haryandika; I Ketut Santra. The Effect of Customer Relationship Management on Customer Satisfaction and Customer Loyalty . **Indonesian Journal of Business and Entrepreneurship (IJBE)**, v. 7, n. 2, p. 139-147, 31 May 2021.

FELDMAN, Ronen; SANGER, James. **The text mining handbook: advanced approaches in analyzing unstructured data**. Cambridge university press, 2007.

GHAZZAWI, Amani; ALHARBI, Basma. Analysis of customer complaints data using data mining techniques. **Procedia Computer Science**, v. 163, p. 62-69, 2019.

HACOHEN-KERNER, Yaakov; MILLER, Daniel; YIGAL, Yair. **The influence of preprocessing on text classification using a bag-of-words representation**. PloS one, v. 15, n. 5, p. e0232525, 2020.

HAN, Jiawei; PEI, Jian; KAMBER, Micheline. **Data mining: concepts and techniques**. Elsevier, 2011.

Hilbe, J.M. (2015). **Practical Guide to Logistic Regression** (1st ed.). Chapman and Hall/CRC..

HOTHO, Andreas; NÜRNBERGER, Andreas; PAAß, Gerhard. A brief survey of text mining. In: **Ldv Forum**. 2005. p. 19-62.

IKONOMAKIS, M.; KOTSIANTIS, Sotiris; TAMPAKAS, V. Text classification using machine learning techniques. WSEAS transactions on computers, v. 4, n. 8, p. 966-974, 2005.

JALILVAND, Mohammad Reza; ESFAHANI, Sharif Shekarchizadeh; SAMIEI, Neda. Electronic word-of-mouth: Challenges and opportunities. **Procedia Computer Science**, v. 3, p. 42-46, 2011.

JOSÉ, I. **KNN (K-Nearest Neighbors) #1**. Disponível em: <https://medium.com/brasil-ai/knn-k-nearest-neighbors-1-e140c82e9c4e>. Acesso em: 02 de Julho de 2023.

KELLER, K.L. Brand Synthesis: The Multidimensionality of Brand Knowledge. *J. Consum. Res.* **2003**, *29*, 595–600

KRISHNA, Gutha Jaya et al. Sentiment Classification of Indian Banks' Customer Complaints. In: **TENCON 2019-2019 IEEE Region 10 Conference (TENCON)**. IEEE, 2019. p. 429-434.

KUMAR, Vineet; REINARTZ, Werner. **Customer relationship management**. Springer-Verlag GmbH Germany, part of Springer Nature 2006, 2012, 2018, 2018.

LAWSON, Richard. **Web scraping with Python**. Packt Publishing Ltd, 2015.

Liu, Y., Wang, Y., & Zhang, J. (2012). **New Machine Learning Algorithm: Random Forest**. Lecture Notes in Computer Science, 246–252.

MERTZ, David. **Text processing in Python**. Addison-Wesley Professional, 2003.

MIRAGEM, Bruno. Novo paradigma tecnológico, mercado de consumo digital e o direito do consumidor. **Revista de Direito do Consumidor**, p. 17-62, 2020.

MIROŃCZUK, M.M. **Information Extraction System for Transforming Unstructured Text Data in Fire Reports into Structured Forms: A Polish Case Study**. *Fire Technol* 56, 545–581 (2020).

MITCHELL, T. M. (1997). **Machine Learning**. New York: McGraw-Hill. ISBN: 978-0-07-042807-2

MOENS, Marie-Francine. **Information extraction: algorithms and prospects in a retrieval context**. Springer Science & Business Media, 2006.

MOLDAGULOVA, Aiman; SULAIMAN, Rosnafisah Bte. **Using KNN algorithm for classification of textual documents.** In: 2017 8th International Conference on Information Technology (ICIT). IEEE, 2017. p. 665-671.

NOGUEIRA, D. **Fé em Deus**. Flavinho Silva. Ao vivo em Cuba. Havana: EMI Music, 2012.

NOORI, Behrooz. Classification of customer reviews using machine learning algorithms. **Applied Artificial Intelligence**, v. 35, n. 8, p. 567-588, 2021.

RUSSELL, Stuart; NORVIG, Peter. **Artificial intelligence: a modern approach**. 2002.

RABBI, B.; RABBI, D. B. K.; GONÇALVES, V. S.; GONÇALVES JÚNIOR, E. R.; BRASIL, J. A. Mineração de dados aplicada a base de reclamações sobre produtos e serviços do programa de proteção e defesa do consumidor / Data mining applied on products and services claims of the consumer protection and defense program. **Brazilian Journal of Development**, *[S. l.]*, v. 4, n. 5, p. 1689–1701, 2018.

RYNGELBLUM, A. L., Vianna, N. W. H., & Rimoli, C. A. (2013). The ways companies really answer consumer complaints. **Marketing Intelligence & Planning**, 31(1), 54–71.

Sperandei S. Understanding logistic regression analysis. Biochem Med (Zagreb). 2014 Feb 15;24(1):12-8. PMID: 24627710; PMCID: PMC3936971.

Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.

SARAWAGI, Sunita et al. **Information extraction**. Foundations and Trends® in Databases, v. 1, n. 3, p. 261-377, 2008.

SAMMUT, Claude; WEBB, Geoffrey I. (Ed.). **Encyclopedia of machine learning**. Springer Science & Business Media, 2011.

SANTOURIDIS, I., & VERAKI, A. (2017). **Customer relationship management and customer satisfaction: the mediating role of relationship quality**. Total Quality Management & Business Excellence, 28(9-10), 1122–1133.

SCHIESSL, José Marcelo. **Descoberta de conhecimento em texto aplicada a um sistema de atendimento ao consumidor**. 2007.

SINGH, Jagdip. Voice, exit, and negative word-of-mouth behaviors: An investigation across three service categories. **Journal of the academy of Marketing Science**, v. 18, p. 1-15, 1990.

SINGH, Sonit. Natural language processing for information extraction. **arXiv preprint arXiv:1807.02383**, 2018.

TADAGOPPULA, S. **Understanding machine learning algorithms — KNN**. Disponível em: <https://medium.datadriveninvestor.com/understanding-machine-learning-algorithms-knn-812840e3e284>. Acesso em: 02 de Julho de 2023.

Uysal, A. K., & Gunal, S. (2014). **The impact of preprocessing on text classification. Information Processing & Management**, 50(1), 104–112.

VERMEER, Susan AM et al. Seeing the wood for the trees: How machine learning can help firms in identifying relevant electronic word-of-mouth in social media. **International Journal of Research in Marketing**, v. 36, n. 3, p. 492-508, 2019.

VIJAYARANI, S. et al. **Preprocessing techniques for text mining-an overview.** International Journal of Computer Science & Communication Networks, v. 5, n. 1, p. 7-16, 2015.

W3Techs. **Usage statistics of client-side programming languages for websites**. [S.I.], 2023 . Disponível em: <https://w3techs.com/technologies/overview/client_side_language>. Acesso em: 28 de Junho de 2023.

ZACH. How to interpret a ROC curve (with examples). **Statology**, 2021. Disponível em: <https://www.statology.org/interpret-roc-curve/>. Acesso em: 10 de Junho de 2023.

ZHENG, Alice; CASARI, Amanda. **Feature engineering for machine learning: principles and techniques for data scientists**. " O'Reilly Media, Inc.", 2018. Cap 3, p. 41-44