Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Análise de Sentimento Multiclasse: uma Abordagem com o uso de Aprendizado de Máquina

Allisfrank dos Santos

Orientadora: Profa. Dra. Heloisa de Arruda Camargo

São Carlos, SP, Brasil

Fevereiro, 2019

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Análise de Sentimento Multiclasse: uma Abordagem com o uso de Aprendizado de Máquina

Allisfrank dos Santos

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Inteligência Artificial.
Orientadora: Profa. Dra. Heloisa de Arruda Camargo

São Carlos, SP, Brasil

Fevereiro, 2019

Dedico este trabalho à minha mãe Socorro e irmã Marquilene, que sempre batalharam para minha formação pessoal, acadêmica e profissional.

Uma hora você têm que tomar uma decisão. As fronteiras não mantêm as pessoas para fora; elas te prendem dentro de si. A vida é confusa mesmo, é assim que fomos feitos.

Então você pode desperdiçar sua vida desenhando linhas ou então você pode viver cruzando-as (...). Se você tem disposição para correr o risco, a vista do outro lado é espetacular. Arrisque-se!!

Grey's Anatomy

Agradecimentos

Primeiramente a Deus, que me permitiu chegar até a fase final deste trabalho, ajudando a não perder as forças mesmo quando elas estavam quase chegando ao fim.

Agradeço a minha família (mãe e irmã), que sempre estiveram ao meu lado apoiando todas ações tomadas até aqui e sempre torceram pelo meu sucesso.

A professora Heloisa, uma orientadora extremamente paciente, que mesmo em momentos bem difíceis, acreditou em mim e não desistiu de me orientar. Sou grato por aceitar me orientar, mesmo sendo o único aluno dela que fugiu de *Fuzzy*, até agora. Como sempre brincamos no laboratório: *Fuzzy* é a cola que nós une.

Agradeço à família CIG (Computational Inteligence Group), Suzane, Jorge, Gerson, Tiago, Léo, Priscila, Mariana Lopes, Mariana Cordeiro, Helano, Eduardo, Fábio, Cris, Maykon, Dino. Uma família que sempre esta disposta a ajudar uns aos outros, meu agradecimentos e desejos de muito sucesso.

Ao Ednei Almeida, minha imensa gratidão, pois esteve presente nos momentos bons e ruins dessa etapa importante. Sempre dando palavras e conselhos de incentivo para a continuação do trabalho.

Aos amigos de Belém, que mesmo de longe sempre fizeram parte dessa jornada, Rafael Guedes,aquele amigo que todos os dias passa pra dar bom dia, a Fabiana Góes, que sempre esteve disposta a me ajudar em todos os momentos e a Luciana Moura, companheira de altas conversas no zap zap, ao amigo Silvio Santiago que sempre torceu pelo meu sucesso.

A cidade de São Carlos, a qual tive amor a primeira vista, e aos amigos que ela me deu: João Batista, Osmair, Alexandre Monte, Adilson, Arieza, Lica, Rodrigo Destefano, Simone Maduenho, Diogo Ferraz.

Aos amigos de república que sempre deixaram as coisas mais leves e divertidas, principalmente aos companheiros Rogério Pompermayer, Marcos Paulo Netto e Luís Carlos Borduchi, com os quais passei bons momentos em São Carlos.

E não posso deixar de fora o Collegium Sapiens, que não foi apenas um local de trabalho por 2 anos, mas também um lugar em que fiz boas amizades.

Meus agradecimentos a UFSCar e a todos os professores e colaboradores do Departamento de Computação pelos ensinamentos passados.

Obrigado a todos!!!

Resumo

No mundo globalizado, a análise de dados, principalmente os textuais, tem se tornado de grande importância para aquisição de conhecimento e informação a partir de dados gerados das mais variadas fontes de dados. Neste aspecto, a internet e as redes sociais compõem a principal base de dados textuais. A Análise de Sentimento é uma forma de mineração de dados na forma de texto, sendo que este tipo de análise visa identificar e/ou analisar a opinião dos usuários sobre uma entidade ou sobre os sentimentos em relação a temas variados. Diversos pesquisadores têm utilizado a Análise de Sentimento para compreender o comportamento dos usuários por meio da polaridade, que podem ser em duas ou três classes. Entretanto, o desafio que se coloca é encontrar meios que ultrapassem a classificação tradicional e conseguir fazer uma análise mais real dos sentimentos expressos, explorando a ideia de análise multiclasse (por meio de classes emocionais). Partindo desses fatos, esse trabalho tem por objetivo estudar aspectos da Análise de Sentimentos com relação ao número de classes de emoções a serem analisadas, bem como a forma de representação dos textos a serem submetidos para classificação. Para isso, foram utilizados algoritmos clássicos de Aprendizado de Máquina (SVM, kNN e Naive Bayes) e uso de técnicas de vetorização como TF - IDF e Word2Vec. Os resultados encontrados mostram que um número reduzido de classes aliado ao uso de Word2Vec como método de representação textual melhoram o resultado da classificação, principalmente com o uso do classificador SVM, obtendo uma acurácia de 58.8% para a base emocional e 68.6% para a base de polaridade.

Palavras-chaves: análise de sentimentos, aprendizado de máquina, polaridade, emoções, classificação.

Abstract

In the globalized world, the analysis of data generated from the most varied sources, especially the textual ones, has become of great importance for the acquisition of knowledge and information. In this respect, the Internet and social networks make up the main textual database. The Sentiment Analysis is a form of data mining in text format, and the purpose of this type of analysis is to identify and / or analyze users' opinions about an entity or about sentiment related to various topics. Several researchers have used the Sentiment Analysis to understand user behavior through polarity, which can be separated into two or three classes. However, the challenge ahead is to find ways that go beyond the traditional classification and achieve a more real analysis of the expressed feelings, exploring t he idea of multiclass analysis (through emotional classes). Based on these facts, this paper aims to study aspects of the Sentiment Analysis related to the number of classes of emotions to be analyzed, as well as the representation form of the texts to be submitted for classification. For this, classic Machine Learning algorithms (SVM, kNN and Naive Bayes) as well as vectorization techniques such as TF - IDF and Word2Vec were used. The results show that a reduced number of classes allied to the use of Word2Vec as a textual representation method improves the classification result, especially with the use of the SVM classifier, obtaining an accuracy of 58.8% for the emotional base and 68.6% for the basis of polarity.

Keywords: sentiment analysis, machine learning, polarity, emotion, classification.

Lista de Ilustrações

Figura 2.1 Representação facial das emoções básicas de Ekman. a) Raiva; b) Medo; c) Desgosto; d) Surpresa; e) Alegria; f) Tristeza. (EKMAN; FRIESEN, 1978) 21

Figura 3.1 Etapas de classificação de texto. (DOSCIATTI, 2015) 34
Figura 3.2 Representação CBOW e Skip-gram, (MIKOLOV et al., 2013). 38
Figura 3.3 Representação espacial Word2Vec. 39

Figura 4.1 Abordagens textual para Análise de Sentimentos. (DOSCIATTI, 2015) 40
Figura 4.2 Representação do SVM, (DOSCIATTI, 2015) 42

Figura A.1 Matriz de dados mxn. 78
Figura A.2 Tipos de Paralelismo. 78
Figura A.3 Diagrama de processo do VHT, (adaptado de (LI, 2014)). 79

Lista de Tabelas

Tabela 2.1 Outras propostas de emoções 20

Tabela 3.1 Exemplo de modelo de anotação. 30
Tabela 3.2 Distribuição das Classes de Emoções no Corpus 31
Tabela 3.3 Distribuição da Polaridade no Corpus 31
Tabela 3.4 Matriz de confusão da concordância entre os anotadores. 32
Tabela 3.5 Matriz de confusão de textos anotados duas vezes pelo anotador 1. . . 33
Tabela 3.6 Matriz de confusão de textos anotados duas vezes pelo anotador 2. . . 33

Tabela 5.1 Distribuição das Classes de Emoções no Corpus após a incorporação . 48
Tabela 5.2 Execução SVM - Emoções 49
Tabela 5.3 Execução SVM - Polaridade 49
Tabela 5.4 Execução Naive Bayes - Emoções 50
Tabela 5.5 Execução Naive Bayes - Polaridade 50
Tabela 5.6 Execução kNN1 - Emoções 51
Tabela 5.7 Execução kNN1 - Polaridade 51
Tabela 5.8 Execução kNN5 - Emoções 51
Tabela 5.9 Execução kNN5 - Polaridade 51
Tabela 5.10 Execução SVM - Emoções - Corpus de Notícias 53
Tabela 5.11 Execução SVM - Polaridade - Corpus de Notícias 53
Tabela 5.12 Execução NB - Emoções - Corpus de Notícias 54
Tabela 5.13 Execução Naive Bayes - Polaridade - Corpus de Notícias 54
Tabela 5.14 Execução kNN1 - Emoções - Corpus de Notícias 54
Tabela 5.15 Execução kNN1 - Polaridade - Corpus de Notícias 55
Tabela 5.16 Execução kNN5 - Emoções - Corpus de Notícias 55
Tabela 5.17 Execução kNN5 - Polaridade - Corpus de Notícias 55
Tabela 5.18 Execução SVM - Polaridade - TAS - PT 56
Tabela 5.19 Execução NB - Polaridade - TAS - PT 56
Tabela 5.20 Execução kNN1 - Polaridade - TAS - PT 56

Tabela 5.21 Execução kNN5 - Polaridade - TAS - PT 57
Tabela 5.22 Classificação de Novos Exemplos - SVM 58
Tabela 5.23 Classificação de Novos Exemplos - NB 59
Tabela 5.24 Classificação de Novos Exemplos - kNN1 59
Tabela 5.25 Classificação de Novos Exemplos - kNN5 60
Tabela 5.26 Inferência de dados com modelo SVM 61
Tabela 5.27 Inferência de dados com modelo NB 62
Tabela 5.28 Inferência de dados com modelo kNN1 63
Tabela 5.29 Inferência de dados com modelo kNN5 63

Lista de Abreviaturas e Siglas

AH	Árvore de Hoeffding
AM	Aprendizado de Máquina
API	Application Programming Interfaces
AS	Análise de Sentimentos
FCD	Fluxo Contínuo de Dados
GBDT	Gradient Boosted Decision Tree
HDFS	Hadoop Dristributed File System
kNN	k-Nearest Neighbor
LH	Limiar de Hoeffding
NB	Naive Bayes
PI	Item de Processamento
PLN	Processamento de Língua Natural
SVM	Support Vector Machine
TF - IDF	Term Frequency – Inverse Document Frequency
VFDT	Very Fast Decision Tree
VHT	Vertical Hoeffding Tree

Sumário

1 Introdução 14
1.1 Contexto 14
1.2 Motivação 16
1.3 Objetivos 17
1.3.1 Objetivo Geral 17
1.3.2 Objetivos Específicos 17
1.4 Organização do Trabalho 18

2 Análise de Sentimentos 19
2.1 Emoções 19
2.2 Análise de Sentimentos e Computação 21
2.3 Análise de Sentimentos e Textos 23
2.4 Trabalhos Recentes na Área de Análise de Sentimentos 24

3 Corpus Anotado para Análise de Sentimentos 26
3.1 Corpus em Análise de Sentimento 26
3.2 Processo de Construção do Corpus 28
3.2.1 Captura dos *Tweets* 29
3.2.2 Processo de Anotação 30
3.3 Processamento de Textos - Classificação de Emoções 33
3.3.1 Pré-Processamento 34
3.3.2 Representação Textual 36

4 Algoritmos de Classificação 40
4.1 Identificação de Sentimentos em Textos - Abordagens 40
4.2 Aprendizado de Máquina 41
4.3 Algoritmos de Aprendizado de Máquina para Classificação 42
4.3.1 Máquina de Vetores de Suporte - SVM 42
4.3.2 Naive Bayes - NB 43
4.3.3 K-Nearest Neighbors - KNN 44
4.3.4 Medidas de Avaliação 45

5 Experimentos e Resultados 47

5.1 Aspectos Avaliados nos Experimentos 47
5.2 Experimentos com a Base Apresentada no Capítulo 3 49
5.2.1 Análise de Sentimento - Outras bases 52
5.3 Classificação de Novos Exemplos 57
5.3.1 Inferência das Emoções do Usuários 60

6 Conclusão 65
6.1 Conclusão 65
6.2 Trabalhos Futuros 66

Referências 67

Apêndices 72

APÊNDICE A Classificação em Flúxo Contínuo de Dados 73
A.1 Árvores de Decisão 74
A.2 Árvore de Hoeffding 75
A.3 Árvore de Decisão Paralela 77
A.3.1 Tipos de Paralelismo 77
A.3.2 Vertical Hoeffding Tree 78
A.4 Experimentos 80

1

Introdução

Neste capítulo será apresentado o contexto no qual esta dissertação de mestrado está inserida, evidenciando as razões que impulsionaram a pesquisa para a temática abordada. Apresenta-se, ainda, os objetivos gerais que nortearam esta pesquisa e, por fim, a estrutura do trabalho.

1.1 Contexto

A percepção acerca das coisas que nos rodeiam está condicionada, em certo grau, à maneira como as demais pessoas veem e avaliam o mundo. Opinar a respeito de alguma entidade, como um filme, um produto, um lugar ou até mesmo sobre outras pessoas, é um processo relevante para as atividades humanas, uma vez que isto exerce influência no comportamento e na tomada de decisão do ser humano.

Ao opinar a respeito de uma entidade nas mídias sociais, o usuário expressa seus sentimentos e emoções. Entende-se como entidade, o alvo de uma opinião, como um filme, uma empresa, uma pessoa. O ato de expressar opinião, cresce a partir do momento em que há maior disponibilidade de conteúdo na *Web* e pelo fato de que os usuários compartilham cada vez mais o que pensam ou sentem a respeito do conteúdo disponibilizado.

O conteúdo gerado por usuários da *Web* cresce de maneira exponencial e segundo (PAK; PAROUBEK, 2010) as plataformas de mídias sociais, como *Twitter*, *Facebook*, *Linkedln*, sites comerciais, permitem que seus usuários compartilhem suas experiências e opiniões sobre os mais variados temas, como política, economia, produtos, pessoas, eventos, entre outros. Esse conteúdo possui um imenso valor em tempo real para avaliação de *marketing*, apoio a tomada de decisão, gerenciamento de desastres, crises globais, análise de pessoas, entre outros.

O *Twitter* é uma das plataformas de mídias sociais mais populares, no que diz respeito a postagens de conteúdo opinativo e emocional. Esta plataforma foi lançada em 2006 e se constitui entre as principais fontes formadoras de opinião, no que tange

as mais diversas entidades (MARTINS, 2014). Empresas dos mais diversos segmentos usam o *Twitter* como meio de monitorar avaliações dos clientes em relação a produtos e conteúdos. Estudos avaliam as reações, satisfações e sentimentos dos usuários a respeito de uma determinada entidade.

De acordo com (BRITO, 2017), o conteúdo gerado em plataformas de mídias sociais produz um grande volume de dados textuais não-estruturados. Diversas técnicas vêm sendo empregadas para estruturar esses dados, de modo que haja compreensão sobre este tipo de informação tornando possível a realização de inferência sobre eles, de acordo com o domínio de aplicação e do que se espera obter como resultado final.

Nesse contexto, surge uma área de pesquisa conhecida como Análise de Sentimentos - AS. A AS, por meio de recursos computacionais, analisa a opinião, sentimentos e subjetividades contidas nos documentos (PANG; LEE et al., 2008). Questões de sentimentos humanos na *Web* tem sido alvo de estudos desde a década de 1990, conforme mostra (PICARD et al., 1995), onde pesquisas já vinham sendo realizadas a respeito das emoções humanas reconhecidas por computador.

Uma das principais questões da AS está na busca de mecanismos que sejam capazes de inferir o real sentimento envolvido em uma opinião expressa por um usuário sobre uma entidade, como um filme de grande repercussão, por exemplo. Por trabalhar com domínios predominantemente textuais, ou seja, em linguagem natural, faz-se necessário o uso de técnicas de Processamento de Língua Natural - PLN, para tratar questões como, normalização dos dados, remoção de *stopwords*, lematização, entre outros, para que, ao final desses processos, os dados estejam dispostos de maneira estruturada para serem processados por técnicas computacionais de Aprendizado de Máquina - AM, por exemplo, na busca pela identificação de sentimentos e emoções presentes nos dados.

Dessa maneira, pesquisas têm sido desenvolvidas com o intuito de combinar técnicas de PLN e AM, de modo a identificar, por exemplo, conteúdo opinativo presente em um texto, a respeito de uma entidade.

A AS consiste em extrair informações de textos em linguagem natural, com o objetivo de obter, de forma automática, a polaridade de um texto ou sentença, dada uma entidade, por meio de recursos computacionais (LIU, 2010). Essa é a forma tradicional de análise, no entanto, trabalhos recentes realizam análises baseadas em sentimentos mais realistas a respeito das entidades, como por exemplo no trabalho de (DOSCIATTI; FERREIRA; PARAISO, 2015), que usa classes emocionais para identificar os sentimentos envolvidos em uma opinião.

Dada a seguinte sentença: *"O novo smartphone, possui um tempo de bateria muito curto"*, isso me deixou bastante decepcionado", pode-se inferir que ela possui um conteúdo de teor negativo, logo sua polaridade é negativa, tendo em vista que fica clara a decepção

com o *"novo smartphone"*, devido ao curto tempo de duração da bateria. Entretanto, se considerarmos uma análise mais realista da sentença, ela poderia ser classificada como sendo uma mensagem de tristeza ou desgosto, baseado no fato do usuário estar decepcionado com a bateria ter pouca durabilidade, esse seria o real sentimento do usuário ao postar essa frase em alguma mídia social. Essas questões serão discutidas ao longo do texto.

1.2 Motivação

Atualmente, a principal tarefa de AS é classificar a polaridade do sentimento (LIU, 2010). Contudo, em relação à expressão de sentimentos humanos, é necessário levar em consideração mais do que a polaridade das emoções (positiva, negativa) ou a ausência de emoção, neutra. Ou seja, é necessário identificar o real sentimento no momento de uma postagem, dado que ao expressar uma opinião, o usuário que expor um sentimento que vai além de positividade ou negatividade. Por trás do texto, há uma satisfação ou insatisfação, uma relação de alegria ou tristeza ou uma relação de amor e ódio, que expressa um sentimento mais realista do quem postou a mensagem e que identificando apenas a polaridade, essas informações mais subjetivas acabam sendo suprimidas e deixando a opinião expressa, menos impessoal e realista.

A área de AS possui diversos cenários de atuação, que permitem melhorar o processo de tomada de decisão, tais como:

- **Análise de pessoas:** é uma análise que identifica o grau de aceitação ou rejeição de figuras públicas (políticos, artistas), em sua maioria, em determinados momentos, o que ajuda na tomada de decisão de como conduzir o andamento de uma campanha política ou a divulgação de um trabalho artístico, por exemplo.

- **Análise de produtos:** a aceitação de um produto no mercado pode ser identificada por meio da AS em mídias sociais. Para uma empresa, entender como as pessoas reagem a um novo produto é importante para determinar a continuidade da sua produção. É possível ainda desenvolver estratégias de venda e divulgação, levando em consideração um novo público alvo ou a remodelação para a imagem do produto ou da própria empresa.

- **Audiência de programas de televisão:** entender o sentimento, aceitação dos expectadores sobre participantes de um programa de televisão e fazer previsões de um possível ganhador, quando a vitória de um participante depender de votos do público. Tendo em vista que, muito se especula a respeito de um possível ganhador, ou até mesmo para provar que o resultado final foi condizente com avaliações anteriores.

Existem diversas atuações possíveis para a tarefa de AS que vão além dessas mencionadas anteriormente (RAVI; RAVI, 2015).

Tendo como referências tais aplicações, e considerando a principal tarefa da AS explorada na maioria dos trabalhos, que é a classificação da polaridade, é importante ressaltar que definir um sentimento é ir além de dizer se ele é positivo, negativo ou neutro. Identificar um sentimento é deixar evidente a real emoção que o usuário quer transmitir ao se expressar por meio de mídias sociais, em sua maioria de maneira textual.

Segundo (ARAÚJO et al., 2013), o volume de dados gerados pelo *Twitter* é imenso, pois é uma plataforma com milhares de usuários ativos e que produzem conteúdo textual 24 horas por dia a respeito dos mais diversos domínios. Então, é necessário técnicas que sejam capazes de lidar com esses dados para extrair deles conteúdo relevante para uma análise mais realista. Assim, neste trabalho a aplicação de técnicas de processamento de texto e AM serão usadas, com o intuito de realizar uma inferência mais realista a respeito dos dados utilizados para análise.

1.3 Objetivos

1.3.1 Objetivo Geral

O objetivo deste trabalho de mestrado e analisar como o número de classes emocionais interferem no desempenho da identificação das emoções dos usuários em postagens textuais e analisar possíveis melhorias na classificação das emoções, por meio do uso representações textuais do tipo *embedding*, com uso de algoritmos AM para realização da tarefa de classificação textual, de textos extraídos do *Twitter* a respeito dos três finalistas do programa de televisão MasterChef Profissionais Brasil 2017. Nesse contexto, a identificação do real sentimento será feita por meio da classificação de *tweets* em classes emocionais (alegria, tristeza, raiva, medo, desgosto, surpresa e neutro), na busca por bons desempenhos (maior precisão na classificação) na tarefa de AS, possibilitando ir além da classificação quanto a polaridade dos sentimentos e evidenciar o real sentimento expresso pelos usuários, que transpassam os limites do positivo, negativo ou neutro. Com o auxílio de técnicas de PLN e algoritmos clássicos de AM como (*Support Vector Machine* - SVM, *Naive Bayes* - NB e *k-Nearest Neighbor* - kNN) para classificação multiclasse de dados textuais que expressem opiniões e sentimentos de usuários.

1.3.2 Objetivos Específicos

- Construção de uma base rotulada para realização dos experimentos, cujo domínio é o MasterChef Profissional Brasil 2017. A coleta dos dados foi realizada por meio da API - (*Application Programming Interface*) disponibilizada pelo *Twitter* para

captura de dados em tempo real, bem como por um período de tempo após a postagem. Após a coleta, alguns *tweets* foram selecionados para rotulação manual em duas categorias de classes: polaridade (positivo, negativo) ou neutro em caso de ausência de polaridade e em classes emocionais (alegria, tristeza, raiva, desgosto, medo, surpresa) ou neutro em caso de ausência de conteúdo emocional. O processo de seleção dos *tweets* candidatos à rotulação será descrito no capítulo de construção da corpus;

- Analisar a opinião dos usuários a respeito dos finalistas da edição do programa, identificando o real sentimento deles sobre cada um dos finalistas.

1.4 Organização do Trabalho

Capítulo 2 - Análise de Sentimentos: apresenta conceitos a respeito da Análise de Sentimento e os principais trabalhos relativos a área.

Capítulo 3 - Corpus Anotado para Análise de Sentimentos: descreve o processo de construção de corpus para a tarefa de Análise de Sentimentos, mais especificamente o corpus criado para este trabalho.

Capítulo 4 - Algoritmos de Classificação: apresenta os algoritmos usados neste projeto, suas aplicações em análise de texto.

Capítulo 5 - Experimentos e Resultados: neste capítulo serão mostrados os experimentos realizados com o corpus construído e bem como experimentos em outras bases de conhecimento.

Capítulo 6 - Conclusões e Trabalhos Futuros: apresenta as considerações finais a respeito dos métodos e resultados obtidos no trabalho.

2

Análise de Sentimentos

Neste capítulo, o conceito de Análise de Sentimentos é apresentado e discutido, por meio da compreensão das emoções a partir do estudo das ciências cognitivas como a filosofia e a psicologia chegando à forma de abordagem do tema no contexto da computação. Emoção possui um conceito desafiador para diversas áreas, que geram discussões e teorias que levaram a construção de modelos estruturais de emoções. Áreas como a Ciência da Computação buscam cada vez mais entender as emoções e aplicar tais modelos estruturais na construção de modelos computacionais que sejam capazes de identificar emoções expressas por usuários *Web*.

Os termos "emoção"e "sentimento"são bastante empregados em trabalhos de AS, e em muitas ocasiões são tratados como um termo único. No entanto, ambos possuem conceitos e formas diferentes de serem empregados. Sentimentos podem ser vistos como a tradução, a resposta as emoções, como o indivíduo se sente diante de uma emoção. Dessa forma, neste trabalho o termo utilizado é o sentimento, uma vez que busca-se traduzir por meio da classificação textual, os sentimentos contidos em textos tendo como base para esses sentimentos, as emoções propostas na literatura, conforme será mostrado ao longo deste capítulo.

2.1 Emoções

A princípio todas as pessoas sabem conceituar emoção, até que alguém pergunte formalmente qual a sua definição (FEHR; RUSSELL, 1984). Diversas serão as respostas na tentativa de conceituar emoção. Entretanto, não há um consenso a respeito da sua definição, pois trata-se de um conceito de natureza subjetiva, e pelo fato do termo ser usado para descrever uma variedade de estados cognitivos e fisiológicos do ser humano (GAZZANIGA; HEATHERTON, 2007). Muitas vezes usamos o termo emoção para descrever desejos biológicos como fome, por exemplo. Essa constatação evidencia que o conceito de emoção também pode ser definido a partir de estados mentais influenciados por sentimentos oriundos de alterações fisiológicas. Dessa forma, conceituar emoção não é uma

tarefa simples de se fazer, tendo em vista que o seu conceito pode ser formulado a partir do olhar, do estado psicológico ou fisiológico de quem vai elaborar esse conceito.

No entanto algumas definições são descritas para conceituar emoção, entre elas está o conceito proposto por (LANG, 1995), que afirma que em termos psicológicos e comportamentais, emoções são vistas como respostas sistêmicas que ocorrem quando ações altamente motivadas são proteladas ou inibidas, dessa forma as emoções dizem respeito à execução de algo importante ao organismo. As emoções podem ser consideradas como movimentos ou ações do corpo, "públicas" no sentido em que são visíveis para terceiros a olho nu.

De um modo geral, as emoções são nossas primeiras respostas às situações cotidianas. São mecanismos básicos que podem incluir alterações em nossa fisiologia, vivências subjetivas e em nosso comportamento.

Diante desses conceitos, diversas teorias foram desenvolvidas em ciências cognitivas, cada uma com seus conceitos de emoção. A partir dessas teorias, três modelos de emoções foram propostos: modelos discretos; modelos dimensionais e os modelos baseados na teoria Appraisal (DOSCIATTI, 2015).

Neste trabalho será abordado apenas o modelo discreto, que tem por objetivo categorizar as emoções partindo do princípio que elas são independentes umas das outras. Um dos principais pesquisadores desse modelo é o psicólogo americano Paul Ekman, que propôs em (EKMAN; FRIESEN, 1978) as seis categorias de emoções básicas ou emoções puras, que são: alegria, tristeza, raiva, medo, desgosto e surpresa. Essas são consideradas emoções básicas, uma vez que é possível provar por meio de experimentos psicofísicos que essas seis emoções podem ser representadas através de expressões faciais universais, ou seja, podem ser compreendidas independente da localidade, língua e cultura, conforme podemos ver na Figura 2.1. Tais emoções foram as usadas neste trabalho, para tratar as questões de AS em modelos computacionais de AM no processo de classificação textual em Português Brasileiro.

Tabela 2.1 – Outras propostas de emoções

Autor	Emoção Fundamental
(MCDOUGALL, 2015)	Raiva, repugnância, alegria, medo e esperança
(WATSON, 1930)	Medo, amor e raiva
(ARNOLD, 1960)	Aversão, coragem, tristeza, raiva, desejo, desespero, medo, ódio, esperança, amor e tristeza
(IZARD, 1971)	Desprezo, raiva, repugnância, angústia, medo, culpa, interesse, alegria, vergonha e surpresa
(MOWRER, 1960)	Dor e prazer

Outros modelos de emoções foram propostos além do modelo de Paul Ekman, como podemos ver na Tabela 2.1.

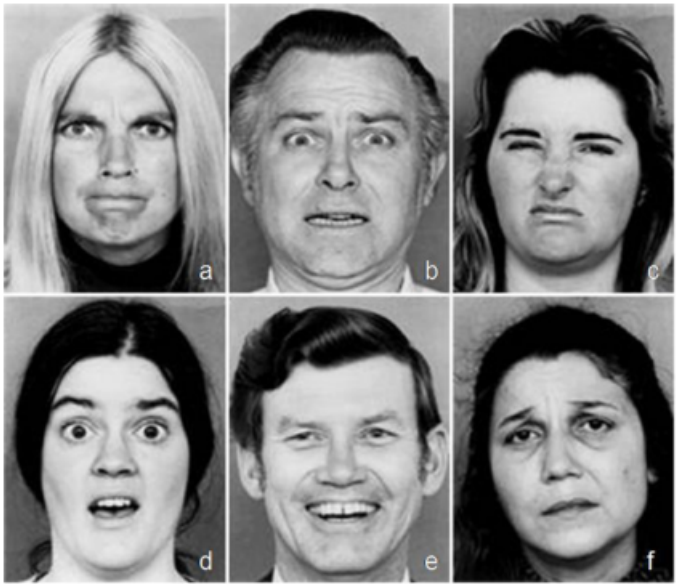


Figura 2.1 – Representação facial das emoções básicas de Ekman. a) Raiva; b) Medo; c) Desgosto; d) Surpresa; e) Alegria; f) Tristeza. (EKMAN; FRIESEN, 1978)

Muitos são os esforços para encontrar uma definição ideal de emoção, entretanto não tem sido uma tarefa fácil para um ser humano identificar as emoções expressas por outro ser humano, pois trabalhos como de (ORTONY; NORMAN; REVELLE, 2005), mostram que a dificuldade na identificação das emoções humanas ocorrem em função das diferenças individuais, das experiências vividas e de fatores genéticos.

A Ciência da Computação tem sua contribuição nos estudos das emoções, principalmente na relação máquina e homem (GROSSMAN; FRIEDER, 2012), onde sistemas computacionais buscam a compreensão das emoções humanas, seu armazenamento, processamento e construção de modelos computacionais das emoções humanas de maneira automática.

Emoções ou afetos em usuários *Web* são vistos como estados identificáveis ou pelo menos processos identificáveis. Com base no estado emocional identificado do usuário, o objetivo é conseguir uma interação tão real ou humana quanto possível, adaptando-se perfeitamente ao estado emocional do usuário e influenciando-o através do uso de várias expressões.

2.2 Análise de Sentimentos e Computação

A emoção é fundamental para as experiências humanas, influenciando a cognição, a percepção e as tarefas cotidianas, como aprendizagem, comunicação e tomadas de decisão. Nesse contexto, estudos computacionais nas áreas de Inteligência Artificial, Interação Humano Computador, Processamento de Língua Natural buscam compreender a interação entre o homem e a máquina, principalmente no campo das emoções humanas e como elas

acontecem a partir do momento em que essas interações tem se intensificado a medida que a máquina ganha maior espaço nas relações pessoais, sociais e de trabalho (PICARD et al., 1995).

A computação afetiva é um campo emergente de pesquisa que visa capacitar sistemas inteligentes a reconhecer, sentir, inferir e interpretar emoções humanas (PORIA et al., 2017).

No campo da Inteligência Artificial, diversas pesquisas estão sendo realizadas para construção de sistemas de detecção automática de emoções principalmente em conteúdo textual, mas também em fala e imagens. (PICARD et al., 1995) foi pioneira em pesquisas computacionais para identificação de emoções em texto. E a partir de então há um crescente interesse por pesquisas nesse campo, a partir do momento em que o uso de mídias sociais torna-se uma fonte de dados textuais rica em conteúdo para identificação de emoções humanas por meio de técnicas computacionais de AM.

Para (LIU, 2012) a AS é a área de estudo que tem por atividade a avaliação de opiniões, de atitudes e emoções das pessoas a respeito de entidades e seus atributos, expressos de maneira textual. Onde normalmente essa avaliação pode ser realizada por meio da classificação de sentimento ou classificação de polaridade dos sentimentos em: positivo, negativo e neutro.

Em (THET; NA; KHOO, 2010) a AS é um tipo de análise de texto, sob a ampla área de PLN e Mineração de Texto, que analisa o sentimento em uma determinada unidade textual com o objetivo de compreender as polaridades das opiniões expressas e os tipos de emoções para vários aspectos de um assunto. Sentimentos, tais como opiniões, atitudes, pensamentos são estados privados de indivíduos que não são abertos a observação ou verificação objetiva. Eles são expressos em linguagem usando expressões subjetivas.

Terminologias diferentes são usadas para conceituar o processo computacional de AS, como por exemplo: Análise de Opinião, Mineração de Opinião, Mineração de Sentimento, Análise de Emoção. No entanto tais expressões, segundo (MEDHAT; HASSAN; KORASHY, 2014) podem ser empregadas de maneira intercalada, possuindo o mesmo significado quando inseridas em um contexto.

Alguns autores discordam quanto a mutualidade dos termos Análise de Sentimento e Mineração de Opinião, (TSYTSARAU; PALPANAS, 2012) afirmam que ambas possuem origens diferentes. Mineração de Opinião surge do contexto de Recuperação de Informação e visa a extração e processamento da opinião dos usuários sobre um produto, filme ou outras entidades. Já a Análise de Sentimento é originada da área de Processamento de Língua Natural e tem como tarefa a recuperação de sentimentos expressos em textos. No entanto, na sua essência são semelhantes e se enquadram no âmbito da análise de subjetividade.

De um modo geral, os termos sentimento, emoção, opinião e afeto são comumente referidos em Análise de Sentimentos, e autores como (MUNEZERO et al., 2014) afirmam que faz-se necessária uma diferenciação entre eles. Sendo assim, sentimentos seriam fenômenos conscientes, focados na pessoa. As emoções seriam expressões sociais dos sentimentos, que podem ser influenciadas pela cultura. Opinião são as interpretações pessoais de informações de uma entidade, podendo ou não conter sentimentos e emoções. O afeto seria o mais abstrato entre eles, uma vez que ele é considerado não consciente para o ser humano e precede sentimentos e emoções. Em (PICARD et al., 1995) não há distinção entre emoção e afeto, uma vez que a emoção é uma das forma de expressar afeto.

Diversos são os conceitos a respeito da definição do que é emoção e o seu processamento computacional, tendo em vista que este tema tem sido abordado nas mais diversas áreas, principalmente: psicologia, linguística e computação. Na atualidade, serviços *Web* geram continuamente um fluxo grande de dados textuais com um alto conteúdo emocional, possibilitando análise sobre os dados, seja de forma: acadêmica, comercial, social, política, econômica, *marketing*, entre outras. Isso tem sido um grande desafio para todas essas áreas, uma vez que busca-se fazer com que sistemas computacionais sejam capazes de reconhecer principalmente a subjetividade emocional expressa na grande maioria dos dados textuais.

2.3 Análise de Sentimentos e Textos

Uma das principais formas de expressão nas mídias sociais é por meio de textos, onde os usuários são fortemente induzidos a opinarem sobre acontecimentos, eventos, produtos, serviços. Com isso, surge a necessidade de análise desses conteúdos textuais para determinar a atitude, emoção, opinião, avaliação ou sentimento do autor da mensagem em relação a uma entidade (PANG; LEE et al., 2008).

E junto dessa necessidade, alguns desafios surgem na tarefa de análise dos fragmentos textuais gerados pelos usuários:

- subjetividade de conteúdo dos textos a serem analisados;
- uso informal da língua de escrita do texto, fazendo uso de gírias, sarcasmo, ironia e ícones de emoção (*emoticons*);
- representação dos dados, de maneira não estruturada;
- variedade de domínios citados em uma mesma mensagem;
- tamanho do texto a ser analisado, limitações de caracteres para escrita do texto a ser analisado.

A tarefa de AS é um problema de classificação de textos, onde a fonte de dados para análise está em língua natural. Um dos principais aspectos que o PLN tem que lidar são os diferentes níveis de análise textual: documento, sentença e aspecto (SERRANO-GUERRERO et al., 2015).

- A análise no nível de documento considera o texto como um todo, sendo uma opinião positiva ou negativa sobre uma entidade (MORAES; VALIATI; NETO, 2013).
- A análise no nível de sentença considera o sentimento expresso em cada frase ou sentença de um documento. O primeiro passo é identificar se a frase é subjetiva ou objetiva. Se for subjetiva, classifica em positiva ou negativa (MEDHAT; HASSAN; KORASHY, 2014).
- A análise no nível de aspecto leva em consideração, quando há a necessidade de uma informação mais precisa a respeito de uma entidade e assim, cada aspecto do texto do classificado de acordo com a polaridade (THET; NA; KHOO, 2010) ou emoções.

Categorizar textos em cada um desses desses níveis, tem uma grande relação com a origem dos dados. A fonte geradora deles é ainda um dos grandes desafios a ser tratado, uma vez que para cada tipo de fonte, teremos um tipo de dado. Por exemplo, em fontes de dados jornalístico o dado a ser a analisado é completamente diferente dos dados encontrados no *Twitter*. Em dados do *Twitter*, é utilizada uma norma mais popular da língua e, portanto, há uma maior preocupação com o processamento do texto, e a análise em sua grande maioria se dá em nível de sentença, devido as limitações de caracteres para escrita. Já em textos jornalísticos, o uso da norma culta é predominante e algumas etapas de processamento podem ser dispensadas (correção de escrita, por exemplo), pode-se fazer uma análise em qualquer nível, já que são dados com longos textos, permitindo uma maior exploração de análise sobre os dados.

Dentro de Análise de Sentimentos em texto, existem inúmeras questões a serem discutidas, principalmente quanto à forma de tratamento do texto para ser usado nessa tarefa, tais como, construção de corpus, preprocessamento textual, normalização entre outros, que serão discutidos no Capítulo 3.

2.4 Trabalhos Recentes na Área de Análise de Sentimentos

Trabalhos mais recentes mostram que as pesquisas continuam com grande impulso na área de Análise de Sentimentos:

Em (SILVA et al., 2018) é proposto o uso de uma arquitetura de rede de aprendizagem profunda para a classificação de emoções em mensagens do *Twitter*, usando o modelo de seis emoções de Ekman: felicidade, tristeza, raiva, medo, desgosto e surpresa.

O corpus usado era composto por *tweets* rotulados, que contém cerca de 2,5 milhões de dados e o uso do modelo preditivo Word2Vec para aprender as relações de cada palavra e transformá-las em números que a rede profunda recebe como entrada. Nossa abordagem alcançou uma precisão de 63% com todas as classes e 77% de precisão em um esquema de classificação binária.

No trabalho de (STOJANOVSKI et al., 2018) apresenta um sistema de aprendizagem profunda para a Análise de Sentimentos e identificação de emoções em mensagens do Twitter. O sistema consiste em uma rede neural convolucional usada para extrair características de dados textuais e um classificador para o qual experimentamos vários algoritmos de classificação diferentes. Foram usadas *embeddings* para treinamento de palavras obtidos por aprendizado não supervisionado em corpora de texto. A avaliação do sistema foi feita por meio de análises de sentimento de três classes com conjuntos de dados fornecidos pela Análise de Sentimento na tarefa do Twitter da competição SemEval. Foi exploramos a eficácia de abordagem para identificação de emoções, usando um conjunto de dados automaticamente anotado com 7 emoções distintas. A arquitetura alcança desempenhos comparáveis a técnicas de ponta no campo da análise de sentimentos e melhora os resultados no campo da identificação de emoções no teste que em que foram usados a proposta como forma de avaliação.

Em (BRUM; NUNES, 2017) apresenta o TweetSentBR, um corpora de sentimentos para o português brasileiro, anotado manualmente com 15.000 sentenças no domínio do programa de TV. As sentenças foram rotuladas em três classes (positiva, neutra e negativa) por sete anotadores, seguindo as diretrizes da literatura para garantir confiabilidade na anotação. Também foram realizados experimentos de linha de base na classificação de polaridade usando três métodos de aprendizado de máquina, atingindo 80,99% na F-Measure e 82.

O trabalho (AHMAD et al., 2018) afirmam que informações na forma de comentários e postagens de sites de microblog podem ser utilizadas por empresas empresas para eliminar as falhas e melhorar os produtos ou serviços de acordo com as necessidades do cliente. No entanto, extrair uma opinião geral de um número impressionante de comentários deu usuários manualmente não podem ser viáveis. Uma solução para isso é usar um método automático para mineração de sentimentos. O algoritmo Support Vector Machine (SVM) é uma das técnicas de classificação amplamente utilizadas para detecção de polaridade a partir de dados textuais. Esta pesquisa propõe uma técnica para ajustar o desempenho do SVM usando o método de busca de grade para análise de sentimento. Neste artigo, três conjuntos de dados são usados para o experimento e o desempenho da técnica proposta é avaliado usando três métricas de recuperação de informação: precisão, recal e f-measure, obtendo resultados satisfatórios.

3

Corpus Anotado para Análise de Sentimentos

Na tarefa de AS o para o Português brasileiro, os recursos linguísticos disponíveis ainda são bastante restritos, principalmente no que se refere a corpus anotado. Diante dessa realidade, e considerando que o processo de análise é baseado nas emoções propostas por (EKMAN; FRIESEN, 1978), neste capítulo será abordado o processo de construção de um corpus anotado para uso no processo de análise de sentimentos em nível de sentença.

O domínio para tratar essa tarefa foi análise de *tweets* do MasterChef Profissionais Brasil 2017. Tal domínio foi escolhido devido a grande interatividade do público a respeito do desempenho dos participantes ao longo do programa, gerando assim um conteúdo textual de cunho emocional nas postagens.

Além do processo de construção do corpus, serão abordados os processos necessários para que o corpus construído seja capaz de ser submetido a análise textual para extração de emoções.

3.1 Corpus em Análise de Sentimento

A AS faz uso de recursos que sejam capazes de fazer com que sistemas computacionais possam processar dados textuais, de modo que suas saídas representem o sentimento do usuário em uma postagem. Um dos principais recursos é o corpus textual anotado, como forma de representação do discurso do usuário.

Um corpus pode ser definido como uma coleção de textos autênticos, legíveis por uma máquina (incluindo transcrições faladas), que são amostras representativas de uma determinada língua natural ou variedade de linguagem, e fornecem uma base material para a construção de Processamento de Língua Natural (INDURKHYA; DAMERAU, 2010).

Na construção de corpus, parâmetros como o Coeficiente *Kappa*, que é um método estatístico para avaliar o nível de concordância entre dois ou mais anotadores, utilizado para medir o grau de relevância desse corpus, que servirá de entrada para treinamento

de modelos computacionais, como por exemplo os de classificação textual e Análise de Sentimentos.

Em Linguística Computacional, esses limites podem variar de acordo com o pesquisador, com o tipo de tarefa que está sendo desenvolvida, uma vez que para tarefas como de Análise de Sentimentos índices menores de *Kappa* são aceitáveis se comparados com atividades de POS tagger, tendo em vista a complexidade e subjetividade da AS. No trabalho de (DOSCIATTI, 2015) o autor diz que um corpus é considerado aceitável quando atingir um Coeficiente *Kappa* superior a 0.67. Já em (ARTSTEIN; POESIO, 2009) o valor de aceitação deve ser superior a 0.8, para uma anotação ser de qualidade e em (EUGENIO; GLASS, 2004) os autores dizem que mais que o valor do Coeficiente *Kappa*, a descrição do processo de anotação, o seu detalhamento, o números de anotações, quais diretrizes foram aplicadas para anotação entre outros fatores, dão condições para avaliar e aceitar um corpus anotado. Obter um grau de concordância considerado aceitável não é uma tarefa trivial quando utiliza-se uma abordagem com seis categorias de emoção, principalmente pelo fato de que em diversas situações não há uma distinção clara dos limites entre algumas emoções como por exemplo raiva e desgosto.

De acordo com (DOSCIATTI; FERREIRA; PARAISO, 2015), a literatura não é vasta para trabalhos que descrevam a construção de corpus em Português Brasileiro para AS bem como o processo de anotação, suas metodologias, os resultados obtidos nesse processo, as medidas de avaliação e o grau de subjetividade entre os anotadores. Porém, alguns trabalhos foram identificados na literatura descrevendo o processo de construção de corpus para o Português, em sua maioria para Análise de Sentimentos com anotação segundo a polaridade (positiva, negativa e neutra), a seguir alguns desses trabalhos são brevemente descritos.

Em (NASCIMENTO et al., 2012) a construção de um corpus de *tweets* a respeito de comentários de três notícias de grande repercussão no momento da coleta foi realizado com auxilio de três anotadores que realizam esse processo de forma manual, construindo-se assim um corpus anotado com 850 tweets sendo 425 tweets positivos e 425 negativos.

No trabalho de (FREITAS et al., 2014) é apresentado o ReLi (REsenha de LIvro), que consiste em 1.600 resenhas de livros anotados manualmente quanto a presença de opinião sobre o livro resenhado e sua polaridade (positiva ou negativa), não foram consideradas opiniões neutras. A anotação ocorreu a nível de frase (polaridade total da frase) e de segmentos de frases (polaridade de trechos de uma frase que contém polaridade). Para esta tarefa, os textos foram anotados por três anotadores, e foram usadas 400 sentenças para medir o grau de concordância entre os anotadores, que ficou em torno de 98%.

O TweetSentBR (BRUM; NUNES, 2017) é um corpus Análise de Sentimentos para o Português brasileiro, composto de 15.000 sentenças no domínio de programas de TV, rotuladas por sete anotadores, de acordo com sua polaridade (positiva, negativa e neutra).

Em (DOSCIATTI; FERREIRA; PARAISO, 2015) um corpus de notícias para Análise de Sentimento foi construído com 2.000 textos jornalísticos, para o Português do Brasil. Os textos foram anotados com as emoções básicas descritas em (EKMAN; FRIESEN, 1978) e a classe neutro em caso de ausência de emoção, obtendo *Kappa* = 0.38.

Um corpus em Inglês foi descrito em (AMAN; SZPAKOWICZ, 2007), contendo 5.205 textos de *blogs*, anotados em nível de sentença, por quatro anotadores. Aos textos foram atribuídos uma das emoções básicas ou categoria "Emoções Mistas"caso o anotador não identificasse uma categoria adequada com o texto, com valor *Kappa* de 0.76. Ainda houve a classificação do texto em emocional e não emocional com um *Kappa* de 0.65 e atribuição de intensidade para as emoções nas categorias alta, baixa ou média obtendo-se um *Kappa* igual a 0.52.

No trabalho de (ALM; ROTH; SPROAT, 2005) foi descrito um corpus em Inglês com 1.580 textos, que foram de contos infantis, anotados em nível de sentença por dois anotadores, obtendo-se um grau de concordância entre 0.24 e 0.51. Nesse processo os textos foram anotados com as seguintes categorias: raiva, repugnância, medo, alegria, tristeza, surpresa positiva ou surpresa negativa.

Diante da necessidade de dados anotados com as seis emoções em Português brasileiro para o domínio descrito anteriormente, um corpus foi construído a partir de dados textuais extraídos da plataforma *Twitter*, durante a exibição do programa MasterChef Brasil, exibido pela Rede Bandeirantes de televisão.

3.2 Processo de Construção do Corpus

O corpus descrito nesta seção, tem por domínio o MasterChef Profissionais Brasil, um programa de competição culinária, composto por 16 participantes que foi exibido de setembro a dezembro de 2017 pela Rede Bandeirantes de televisão.

A motivação principal para a construção do corpus foi a sua aplicação em um processo de Análise de Sentimentos multiclasse, usando classificadores de AM clássicos. E devido a carência de corpus anotado com as seis emoções básicas para o Português, fez-se necessário a construção de um corpus anotado que atendesse essa necessidade.

A escolha de *tweets* do MasterChef como objeto para construção do corpus se se deu pelo fato de que o programa possui uma grande audiência e com uma grande interação do público por meio do *Twitter*, gerando assim uma grande quantidade de conteúdo textual para análise.

3.2.1 Captura dos *Tweets*

O *Twitter* é uma plataforma de mídia social que foi lançada em 2006, e desde então tem se tornado uma das mídias sociais mais influentes da internet (THELWALL; BUCKLEY; PALTOGLOU, 2011), que permite que seus usuários descrevam seus status por meio de mensagens de textos curtos.

Inúmeros *tweets* são publicados por dia, esse grande volume de dados, acontecimentos importantes que ganham grande notoriedade nas redes sociais e a rapidez com que eles são gerados e propagados, fazem com que a plataforma seja uma das melhores fontes de transmissão de dados e informações públicas em tempo real. Esta foi a principal razão pela qual essa plataforma foi escolhida para a obtenção de dados para a construção do corpus e a realização deste trabalho.

Os *tweets* são disponibilizados por meio de uma API (*Application Programming Intefaces*), que podem ser capturados em tempo real (*Streaming* API) ou em um momento posterior a sua publicação, respeitando um intervalo de tempo determinado pela plataforma.

Para ter acesso ao *Streaming* API é necessário ter uma conta ativa no *Twitter*, criar a aplicação e então códigos de acesso são gerados para que a API funcione. Uma vez criada a aplicação, um *script* em *python* foi desenvolvido para vincular a conta do *Twitter* com a aplicação por meio dos códigos de acesso gerados. A partir de então a captura pode ser realizada, usando um filtro de restrição de busca. Para o corpus em questão, utilizou-se a *hashtag* oficial do programa, "#masterchefbr".

A captura dos *tweets* foi realizada ao longo de todos os episódios da edição MasterChef Profissionais, sempre do início do episódio até uma hora após o seu término, pois o objetivo é classificar o sentimento dos usuários a respeito dos participantes, e muitos comentários emocionais podem ser identificados após o fim do episódio, onde os usuários opinam principalmente a respeito do eliminado do dia.

Ao final do processo de captura de dados, obteve-se 14 arquivos com *tweets* correspondentes a cada episódio, com uma média de 50.000 *tweets* cada um. A fim de reduzir a dimensionalidade dos dados a serem anotados devido a restrições de recursos, como o número limitado de anotadores, algumas medidas foram adotadas. A primeira medida foi definir que os *tweets* a serem anotados se referissem aos finalistas do programa, pois seria possível acompanhar a trajetória do participante ao longo do programa, e ter a percepção do sentimento expresso pelos usuários em relação ao participante desde o início até o final da edição do programa. A segunda medida foi o uso de um filtro, usando como palavra-chave o nome do participante finalista, para selecionar os *tweets* candidatos para anotação.

3.2.2 Processo de Anotação

Terminado o processo de captura e estabelecido os critérios de escolha dos *tweets* a serem anotados, selecionou-se um total de 2.550 textos a respeito dos três finalistas da edição. Os textos estão distribuídos de forma igual para cada um dos participantes e foi selecionada uma amostra de dados de cada episódio, para que assim fosse possível ter dados opinativos do início ao fim da participação do finalista na edição do programa, e poder acompanhar a evolução de sentimento a respeito do participante, de acordo com o seu desempenho nas provas e as avaliações dos jurados.

O processo de anotação foi realizada por três anotadores voluntários. Cada texto foi anotado por dois anotadores diferentes, caso houvesse discordância, um terceiro anotador faria o processo de desempate e elegeria uma classe para o texto.

Foram realizadas três rodadas de anotações onde, na primeira e na segunda rodada foram rotulados 1.000 *tweets* em cada uma delas e na terceira rodada mais 550, somando um total de 2.550 *tweets* rotulados. Esse processo de anotação durou cerca de 3 meses até a sua conclusão.

O processo de anotar consistia em ler os *tweets*, identificar a presença de uma ou mais emoções. Caso houvesse mais de uma, identificar a emoção predominante (alegria, surpresa, medo, tristeza, desgosto, raiva), atribuir uma intensidade a emoção predominante (alta, média ou baixa) e uma polaridade (positiva, negativa). Em caso de ausência de emoção ou polaridade no texto, os anotadores classificariam o *tweet* como neutro.

Para auxiliar os anotadores no processo de anotação, foi fornecido um arquivo com 30 *tweets* previamente rotulados, para servir de exemplo de modelo para anotação, conforme mostra na Tabela 3.1, juntamente com esse arquivo, instruções de como a anotação deveria ocorrer e uma lista com palavras emocionais distribuídas entre as seis emoções básicas extraído do trabalho de (DOSCIATTI; FERREIRA; PARAISO, 2015). O objetivo desse conjunto de instruções era de nivelar o conhecimento dos anotadores, na tentativa de diminuir a subjetividade na tarefa de anotação. Uma rodada de anotação coletiva foi feita para que possíveis dúvidas fossem solucionadas antes das rodadas oficiais.

Tabela 3.1 – Exemplo de modelo de anotação.

Texto	***Emoção por Sentença***	***Emoção Predominante***	***Intensidade***	***Polaridade***
minha maior tristeza nao eh q a Mirna partiu mas Francisco e Clecio ficaram #MasterChefBR	Tristeza	Tristeza	Alta	Negativa
Eu rio de tudo que Pablo diz que é muito engraçado #MasterChefBR	Alegria	Alegria	Alta	Positiva
Estou muito triste com a saída do Ravi. Mas estou feliz com o seu desempenho no programa.	Tristeza/Alegria	Tristeza	Média	Negativa

Finalizada a anotação pelos dois anotadores, foi feita a verificação dos *tweets* discordantes entre eles, para serem analisados por um terceiro anotador para que fosse decidido o rótulo dos casos discordantes. Após esse processo, chegou-se a seguinte distribuição das classes emocionais dentro do corpus conforme mostra a Tabela 3.2.

Tabela 3.2 – Distribuição das Classes de Emoções no Corpus

Emoção	**Total**	**Porcentagem**
Alegria	855	35.5%
Tristeza	101	4%
Raiva	298	11.7%
Medo	90	3.5%
Desgosto	413	16.2%
Surpresa	197	7.7%
Neutro	596	23.2%

Já no contexto da polaridade obteve-se os resultados mostrados na tabela 3.3.

Tabela 3.3 – Distribuição da Polaridade no Corpus

Polaridade	**Total**	**Porcentagem**
Positivo	984	38.6%
Negativo	975	38.2%
Neutro	591	23.2%

Terminado todo esse processo, o passo seguinte foi a extração de medidas de avaliação do corpus anotado, para garantir a confiabilidade da anotação.

Percebe-se que há um grande desbalanceamento entre as emoções, como pode ser observado entre as emoções alegria e medo. Muito disso é por conta da sobreposição de conceitos entre algumas classes de emoções (desgosto e raiva, por exemplo), que acaba dificultando o entendimento e o discernimento do anotador, tendo em vista que o contexto que motivou a escrita da postagem possa não ser compreendido pelo anotador no momento da leitura. Como por exemplo: "não acredito", pode ser uma postagem logo após a eliminação de um participante, deixando evidente que o autor do texto queria expressar surpresa diante da eliminação. No entanto, para o anotador pode soar como uma frase neutra, podendo se questionar: "não acredita em que? Não acredita em algo que foi dito?". Esse tipo de entendimento pode comprometer a rotulação, gerando em muitos casos um desbalanceamento, pois a anotação pode tender a uma dada emoção, justamente por esse tipo situação.

Técnicas como subamostragem e sobreamostragem podem ser utilizadas para lidar com esse tipo de problema. Sendo que a subamostragem tem por finalidade a redução do tamanho dos dados de classe majoritárias, por meio de uma seleção randômica dos dados,

enquanto que a sobreamostragem atua no sentido de aumentar a quantidade de dados com classe minoritárias.

No entanto o uso dessas técnicas comprometem os dados, pois a sobreamostragem produz um aumento nos dados, entretanto não gera nenhum ganho de informação, contribuindo assim para *overfitting* do modelo a ser gerado, uma vez que trabalha na replicação de dados apenas. Já na técnica de subamostragem, há uma redução na quantidade de dados, o que pode gerar perda de informação no processo de descarte de dados.

3.2.2.1 Grau de Concordância entre Anotadores

Uma vez que o processo de anotação foi finalizado, é necessário validar essa tarefa, e para isso faz-se uso de uma medida para avaliar a confiabilidade da concordância entre os anotadores do corpus, antes de ser submetido a algum processamento algorítmico. Essa validação tem a finalidade de atestar que esse corpus é adequado para testar e avaliar a saída de um processamento computacional.

Diversas medidas podem ser usadas para avaliar a confiabilidade da concordância. A escolha depende dos dados, dos recursos utilizados no processo de anotação e do que se espera como resultado. Métodos como Coeficiente de Correlação de *Pearson* e Coeficiente *Kappa* podem ser usados para garantir a confiabilidade de um corpus. Nesse trabalho, o Coeficiente *Kappa* foi usado como medida de confiabilidade entre os anotadores.

Alguns experimentos foram realizados após a finalização do corpus. O primeiro foi para encontrar o grau de concordância entre os anotadores, chegando-se a uma valor *Kappa* de 0.42 entre os dois anotadores, um valor abaixo do indicado pela literatura de linguística de corpus. A Tabela 3.2 mostra a matriz de confusão da concordância entre os anotadores.

Tabela 3.4 – Matriz de confusão da concordância entre os anotadores.

		Anotador 2					
Anotador 1	Alegria	Tristeza	Raiva	Medo	Desgosto	Surpresa	Neutro
Alegria	656	8	7	7	20	20	135
Tristeza	14	48	3	12	12	7	15
Raiva	17	6	159	3	216	9	81
Medo	8	7	0	38	7	2	45
Desgosto	18	6	34	5	114	7	65
Surpresa	46	9	10	6	28	67	75
Neutro	111	7	10	10	30	25	305

Um outro experimento realizado foi para verificar diferenças entre duas anotações quando um texto é rotulado pelo mesmo anotador em momentos diferentes, evidenciando o grau de subjetividade do anotador no processo de rotulação. Foram 200 *tweets* anotados duas vezes, sem que o anotador soubesse quais eram os *tweets* duplicados. As Tabelas 3.3 e 3.4 mostram as matrizes de confusão de textos anotados duas vezes pelo anotador.

Tabela 3.5 – Matriz de confusão de textos anotados duas vezes pelo anotador 1.

		2ª Vez					
1ª Vez	Emoção	Alegria	Tristeza	Raiva	Medo	Desgosto	Surpresa	Neutro
	Alegria	45	0	3	1	2	3	10
	Tristeza	0	10	0	1	0	0	0
	Raiva	0	2	21	0	6	0	0
	Medo	0	1	0	6	1	1	1
	Desgosto	1	0	9	0	10	1	1
	Surpresa	5	1	4	1	1	16	2
	Neutro	8	0	2	1	4	1	18

Tabela 3.6 – Matriz de confusão de textos anotados duas vezes pelo anotador 2.

		2ª Vez						
1ª Vez	Emoção	Alegria	Tristeza	Raiva	Medo	Desgosto	Surpresa	Neutro
	Alegria	52	0	2	0	3	0	5
	Tristeza	0	5	1	0	1	0	0
	Raiva	2	2	5	0	3	0	1
	Medo	2	0	0	8	0	0	0
	Desgosto	1	4	2	2	26	3	3
	Surpresa	1	1	1	0	2	7	3
	Neutro	6	2	1	1	3	5	34

O Coeficiente *Kappa* encontrado para o anotador 1 foi de 0.54 e para o anotador 2 foi de 0.59. Esses resultados deixam evidente que o processo de anotação é uma tarefa subjetiva, sendo um fator que contribui para o baixo valor *Kappa* de um corpus, já que não há uma padronização no processo de anotação de corpus emocional multiclasse.

O mesmo processo foi realizado para medir o grau de concordância entre os anotadores na tarefa de rotular os dados segundo a sua polaridade, chegando a um Coeficiente *Kappa* igual a 0.40.

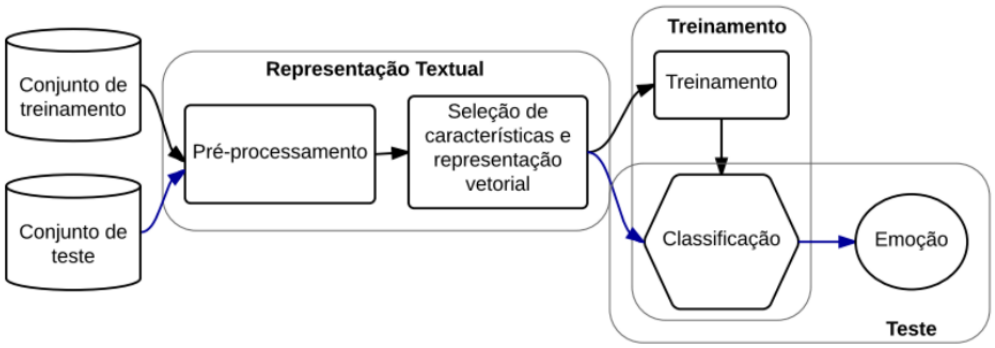
Após todo o processo de anotação e extração de métricas de concordância entre os anotadores, foram realizados alguns processamentos necessários para que o corpus fosse submetido a um processo de extração de conhecimento, como por exemplo, usando técnicas de AM. Esse processo pode variar de acordo com o objetivo da análise a ser realizada. Para esta pesquisa, foram realizadas as etapas descritas a seguir.

3.3 Processamento de Textos - Classificação de Emoções

Antes de submeter o copus descrito na seção anterior a algum processamento de análise computacional, é necessário preparar esse texto para que ele possa ser legível por uma máquina. Para isso algumas etapas devem ser seguidas, conforme mostra a figura 3.1. Essas etapas não são as únicas a serem seguidas, para cada contexto de aplicação do corpus podem haver mais etapas a serem seguidas além das descritas neste trabalho.

Nas próximas subseções será descrita a etapa de representação textual, que consiste na tarefa de pré-processamento e seleção de características. As demais partes, como a fase de treinamento e teste são explicadas no Capítulo 5, que trata dos experimentos e resultados.

Figura 3.1 – Etapas de classificação de texto. (DOSCIATTI, 2015)



3.3.1 Pré-Processamento

Após a coleta do corpus, ele deve estar preparado para que seja processado por técnicas de extração de conhecimento. Nesta fase é realizada a filtragem, limpeza dos dados, exclusão de informações não essenciais para obtenção do conhecimento e representação estruturada dos textos, na forma atributo-valor.

Esta fase é composta por um conjunto de etapas que realizam a adaptação do corpus para o objetivo final, que é a extração de termos candidatos (BENABDALLAH; ABDERRAHIM; ABDERRAHIM, 2017).

Essa etapa do processo de extração do conhecimento é bastante custosa do ponto de vista computacional, uma vez que não existe uma técnica que seja adequada a todos os domínios de aplicações, assim, deve-se ter um bom conhecimento dos dados, da sua origem e de como eles estão dispostos na base de conhecimento, para assim poder determinar quais processes devem ser aplicados para tratamento dos dados.

Algumas técnicas são aplicadas nessa fase, que auxiliam no pré-processamento:

- ***Tokenização*:** *tokens* são os termos, palavras ou expressões compostas, extraídas do texto. Os *tokens* podem ser representados por n-gramas, que são os conjuntos de n termos em sequência. Por padrão, os *tokens* são palavras e possuem delimitadores que auxiliam na sua identificação. Geralmente esses delimitadores são os espaços entre os termos, sinais de pontuação, conforme pode ser visto no exemplo a seguir.

"Amanhã fará sol em São Carlos!"

[Amanhã] [fará] [sol] [em] [São] [Carlos] [!]

No processo de *tokenização*, o "espaço" é desconsiderado, conforme a transformação acima, mas em algumas línguas o espaço não é usado para delimitar os *tokens*, como por exemplo no Japonês, Chinês, Árabe, uma vez que essas línguas não usam o espaço como separadores de termos.

- **Remoção de *stopwords*:** é a exclusão de *tokens* que não possuem um valor semântico de maneira isolada, apenas na compreensão global do texto.

  As *stopwords* são definidadas em uma lista chamada de *stoplist*, lista de palavras que agregam pouco valor no momento da análise. Preposições, conjunções, pronomes, artigos, pontuações de uma língua é que compõem essa lista, no entanto, podem haver outras *stopwords* na composição dessa lista, isso vai depender do domínio de trabalho.

- ***Normalização*:** é a técnica que visa melhorar a qualidade dos textos extraídos da *web*, uma vez que, os textos que originalmente são coletados de redes sociais, geralmente não seguem as normas culta da língua em que são postadas. Havendo a necessidade de aproximar o máximo possível esses textos da norma padrão, no nosso caso o Português do Brasil.

Um *script* em Python foi escrito para realizar a remoção de *stopwords*, caracteres especiais (@, emotions, pontuações, url, #).

Para a tarefa de normalização, foi utilizado o normalizador proposto por (BERTAGLIA; NUNES, 2016), chamado de Enelvo, cujo objetivo é tratar textos da *web* com ruídos (ditos fora do padrão culto), especificamente para o Português e encontrar palavras candidatas para a sua substituição.

Para a *tokenização* um outro *script* Python foi usado nessa tarefa. Os *tokens* foram compostos por unigramas.

Para exemplificar cada um desses passos, um texto extraído do corpus será mostrado, passando por cada uma das etapas descritas anteriormente.

Dado o texto a seguir:

"*@masterchefbr #MasterChefBR, Paola eh uma mal educada, deboxou Irina desde q a moca começou a falar... https://t.co/rPflMFyQXR*".

Após passar a sentença pelo normalizador:

"*@masterchefbr #masterchefbr , paola é uma mal educada , debochou irina desde que a moça começou a falar... https://t.co/rpflmfyqxr*".

Aplicando o *script* de remoção de *stopwords*:

"*masterchefbr paola mal educada debochou irina desde moca comecou falar*"

Resultado da etapa de *tokenização* da sentença

"*['masterchefbr', 'paola', 'mal', 'educada', 'debochou', 'irina', 'desde', 'moca', 'comecou', 'falar']*"

Da sentença original até a sentença final, mostrando os *tokens* obtidos, o texto sofreu modificações, passando por uma correção ortográfica, remoção de hiperlinks, #, @, acentuação e pontuação.

3.3.2 Representação Textual

Os dados obtidos no corpus, são dados textuais representados em língua natural, representação de fácil compreensão humana. No entanto, para que uma máquina possa reconhecer esse corpus como uma entrada válida para extração de conhecimento, é necessário passar esses dados de forma estruturada de modo que os possa ser reconhecido e processado pelos algoritmos.

A representação textual trabalha na extração de *features* e considera-se uma forma de reduzir dimensionalmente o tamanho das entradas a serem processadas, onde geralmente há uma quantidade grande de dados de entrada e pouca informação. Nesse etapa, busca-se extrair os atributos mais relevantes para serem analisados.

3.3.2.1 Presença de Termos e Frequência de Termos

A quantificação de termos em um documento é uma das questões centrais em processamento de texto. As palavras contidas em um documento possuem importâncias diferentes umas das outras. Geralmente substantivos e verbos possuem mais relevância que outros termos como artigos, por exemplo.

Term Frequency - Inverse Document Frequency (TF-IDF), determina quais palavras em um corpus de documentos podem ser mais favoráveis para uso. O TF-IDF calcula valores para cada palavra de um documento por meio de proporção inversa da frequência da palavra em um determinado documento para a percentagem de documentos que a palavra aparece (RAMOS et al., 2003). Palavras com números altos de TF-IDF implicam uma forte relação com o documento em que aparecem, sugerindo que, se essa palavra aparecesse em um consulta, o documento poderia ser de interesse para análise.

- ***TF - Term Frequency**:* mede a frequência com que um termo ocorre em um documento. Como cada documento é diferente em tamanho, é possível que um termo apareça muito mais vezes em documentos longos do que em documentos mais curtos. Assim, o termo frequência é frequentemente dividido pelo comprimento do documento (também conhecido como número total de termos no documento) como forma de normalização, como mostrado na Equação 3.1.

$$\text{TF(t)} = \frac{\text{(Número de vezes que o termo } t \text{ aparece em um documento)}}{\text{(Número total termos do documento)}} \tag{3.1}$$

- ***IDF - Inverse Document Frequency**:* mede a importância de um termo. Enquanto computa TF, todos os termos são considerados igualmente importantes. No entanto, sabe-se que certos termos, como "é", "de"e "aquilo", podem aparecer muitas vezes, mas têm pouca importância. Assim, precisamos pesar os termos frequentes enquanto aumentamos os mais raros, conforme mostrado na Equação 3.2.

$$\text{IDF(t)} = \ln \frac{\text{(Número total de documentos)}}{\text{(Número de documentos com o termo } t \text{ nele)}} \tag{3.2}$$

Dessa forma, o cálculo final segue conforme mostrado na Equação 3.3.

$$\text{TF-IDF} = \text{TF} * \text{IDF} \tag{3.3}$$

3.3.2.2 *Word Embeddings*

Quando fala-se em representação de palavras, primeiramente pensa-se em formas de como converter palavras para uma representação que seja entendível por uma máquina. Um outro pensamento bem recorrente ao falar de representação de palavras, é a redução da dimensionalidade do texto a ser analisado e torná-lo o mais representativo possível, de modo a extrair informações relevantes para o processo de aquisição de conhecimento.

Técnicas de Processamento de Língua Natural junto com aprendizado de máquina, buscam cada vez mais aprimorar esse processo, de modo encontrar métodos de representação de palavras mais representativos e com dimensões cada vez menores. Nesse contexto, trabalhos como o de (BENGIO et al., 2003) propuseram o uso de redes neurais na aquisição de vetores de palavras, as chamadas *word embeddings*, que originalmente foram chamadas de representação distribuída, que tem seu aprendizado com base no uso de palavras. Isso permite que palavras usadas de maneira semelhante resultem em representações semelhantes, capturando naturalmente seu significado. Isso pode ser contrastado com a representação nítida mas frágil em um modelo de saco de palavras onde, a menos que explicitamente gerenciado, palavras diferentes têm representações diferentes, independentemente de como são usadas.

Word2Vec

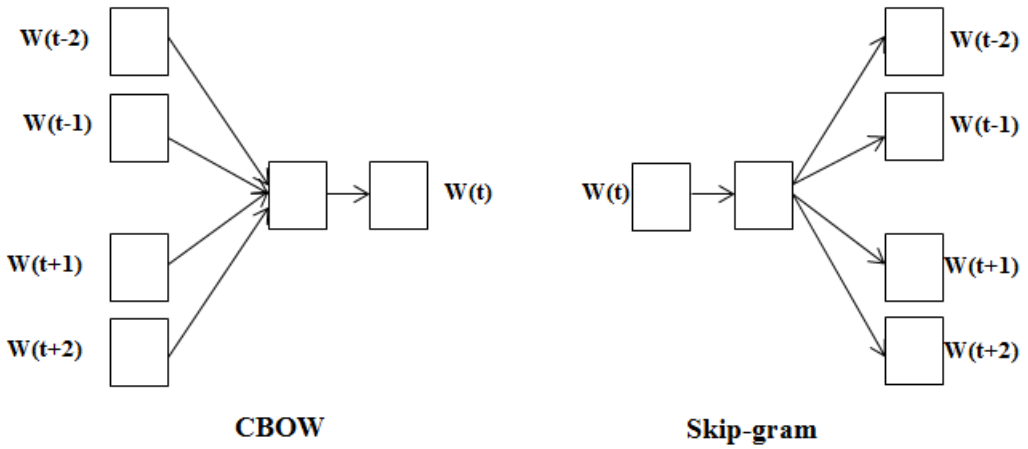
O *Word2Vec* é um modelo de representação de *embeddings*, descrito em (MIKOLOV et al., 2013), eficiente em termos de computação para o aprendizado de incorporação de palavras a partir de texto bruto. A ideia é transformar cada palavra de uma sentença do corpus, em um vetor que a represente numericamente.

O objetivo do *Word2Vec* é chegar em uma representação vetorial de forma não supervisionada a partir das palavras presentes em um texto. Para isso as palavras são associadas a um vetor o qual é denominado *Wordvec*. Com essas associações espera-se que palavras inseridas em contextos similares em uma coleção de documentos tenham vetores próximos em um espaço *"n"* dimensional.

O *Word2Vec* faz parte de uma classe de modelos neurais e pode ser construído a partir de dois modelos de redes neurais:

- ***Continuous bag-of-words - CBOW*:** que tem por objetivo prever a palavra do meio a partir de determinados contextos adjacentes. Para isso uma rede neural é empregada, e sua saída será a palavra procurada, a arquitetura do modelo é mostrada na Figura 3.2.
- ***Skip-gram*:** nessa abordagem o ponto de partida é uma dada palavra, e o objetivo é prever o contexto que ela está inserida. O modelo é mostrado na Figura 3.2

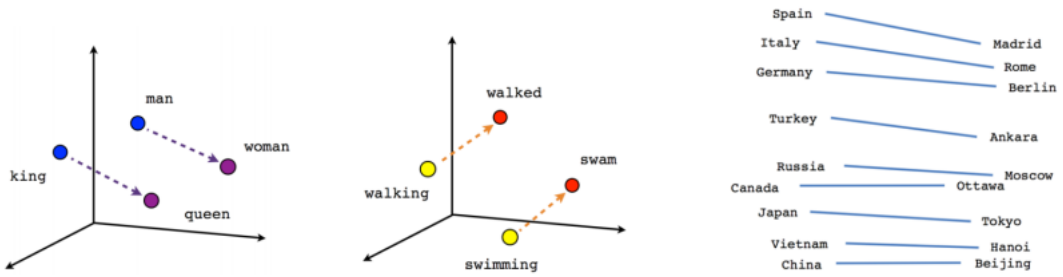
Figura 3.2 – Representação CBOW e Skip-gram, (MIKOLOV et al., 2013).



A escolha de qual modelo usar, pode parecer arbitrária, mas estatisticamente ela tem o efeito de que o CBOW suaviza muitas das informações de distribuição (tratando todo um contexto como uma observação). Na maior parte, isso acaba sendo útil para conjuntos de dados menores. No entanto, Skip-gram trata cada par de contexto-alvo como uma nova observação, e isso tende a melhorar quando o conjunto de é dados grande.

Com dados e contextos suficientes, o *Word2Vec* pode fazer suposições altamente precisas sobre o significado de uma palavra com base em aparências anteriores. Esses palpites podem ser usados para estabelecer a associação de uma palavra com outras palavras (por exemplo, “homem” significa “menino”, “mulher” significa “menina”), ou agrupar documentos e classificá-los por tópicos, ver Figura 3.3. Esses *clusters* podem formar a base da pesquisa, análise de sentimentos e recomendações em campos tão diversos como pesquisa científica, descoberta legal, comércio eletrônico e gerenciamento de relacionamento com o cliente.

Figura 3.3 – Representação espacial Word2Vec.



A saída da rede neural *Word2Vec* é um vocabulário no qual cada item tem um vetor anexado a ele, que pode ser alimentado em uma rede de aprendizagem profunda ou simplesmente consultado para detectar relações entre palavras.

Neste trabalho, foram utilizadas as técnicas TF-IDF e *Word2Vec* com CBOW como forma de vetorização dos textos processados, com o objetivo de demonstrar o desempenho dos algoritmos com as duas técnicas apresentadas. No Capítulo 5 serão apresentados os experimentos e resultados e cada experimento realizado foram usados textos com as duas técnicas.

4

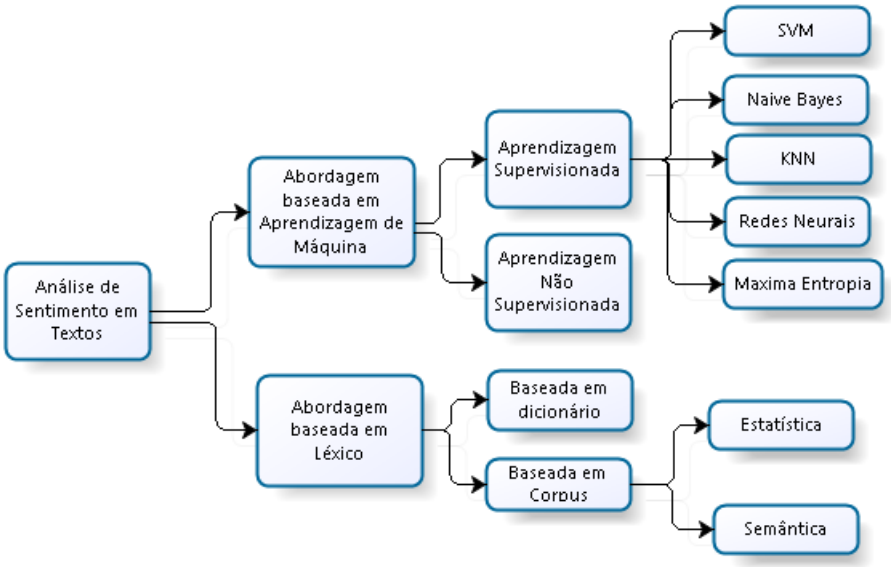
Algoritmos de Classificação

Nesse capítulo serão apresentados alguns conceitos básicos do contexto de AM, e os algoritmos que foram usados para realizar os experimentos deste trabalho e as principais métricas de avaliação para tratar problemas de classificação.

4.1 Identificação de Sentimentos em Textos - Abordagens

Em AS existem duas grandes possibilidades de realizar essa tarefa, uma usando abordagem de AM e outra usando léxico, conforme podemos verificar na Figura 4.1.

Figura 4.1 – Abordagens textual para Análise de Sentimentos. (DOSCIATTI, 2015)



.

A abordagem baseada em AM pode ser dividida em duas partes, uma em que necessita de um grande número de dados previamente rotulados, que é a abordagem supervisionada e quando não existe essa grande disponibilidade de dados rotulados, faz-se uso do aprendizado não supervisionado.

Abordagens baseadas em léxico fazem uso de léxico de termos emocionais. Para (JURAFSKY; MARTIN, 2009) um léxico pode ser entendido como uma estrutura alta-

mente sistemática que define o significado das palavras e como elas podem ser usadas. Para (SPECIA; NUNES, 2004), os léxicos computacionais são recursos criados, geralmente, de forma manual, especificamente para o tratamento computacional.

Alguns trabalhos fazem uso dessas técnicas de formas separadas ou juntas, tais como: (ARAÚJO; GONÇALVES; BENEVENUTO, 2013), que usa abordagem baseada em léxico, (REIS et al., 2015) usa ambas as abordagens, (FOUAD; GHARIB; MASHAT, 2018), que faz uso de abordagem de AM.

Na seção a seguir, será explicado melhor como funcionam os tipos de AM que foi a técnica usada nesta pesquisa e os algoritmos usados neste trabalho.

4.2 Aprendizado de Máquina

No contexto de Análise de Sentimentos, a aprendizagem de máquina segundo (MEULEMAN; SCHERER, 2013) também pode ser chamada de reconhecimento de padrões ou mineração de texto.

Em AM, utiliza-se algoritmos que recebem dados e geram modelos, e a partir de então, conseguem realizar a inferência de novas entradas de dados a partir do modelo treinado.

Existem três tipos de abordagem de aprendizado.

- Aprendizado Supervisionado: nessa abordagem, a inferência de novos dados é realizada por meio de um modelo previamente treinado, onde os dados de treinamento possuem o atributo que contêm o rótulo do dado, ou seja, possui a informação necessária para classificar corretamente o dado (Classe), informação essa que pode ser valores nominais ou contínuos. Esse tipo de aprendizado é aplicado em tarefas de classificação e regressão.

- Aprendizado Não Supervisionado: nesse caso o processo de inferência não possui dados com rótulos previamente definidos, dessa forma busca-se por padrões nos dados que os tornam semelhantes e com isso a possibilidade de relacioná-los afim de inferir informações válidas. Tarefas como a de *Clustering* normalmente parte do aprendizado não supervisionado para identificar grupos semelhantes no conjunto de dados.

- Aprendizado Semissupervisionado: parte do princípio que em um mesmo conjunto de dados disponível para treinamento existem dados rotulados e não rotulados para realizar a inferência de conhecimento, aplicando-se técnica de aprendizado supervisionado ou não supervisionado.

A análise de sentimentos realizada pela técnica de aprendizagem de máquina supervisionada utiliza textos já classificados que servem como base de treinamento. Com o uso de modelos já treinados, tenta-se prever o conteúdo emocional de um texto desconhecido. Uma das grandes dificuldades no uso de aprendizado supervisionado na tarefa de análise de sentimento é a necessidade de uma grande quantidade de textos rotulados para que possa ser garantida a confiabilidade da base de dados.

4.3 Algoritmos de Aprendizado de Máquina para Classificação

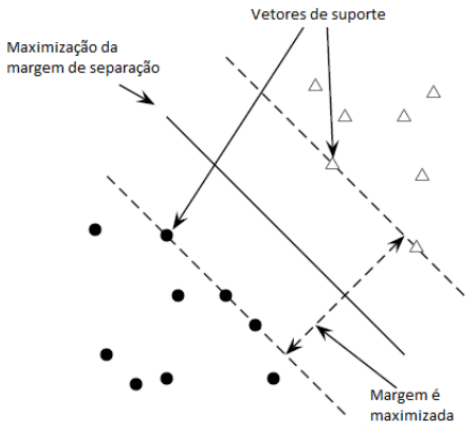
4.3.1 Máquina de Vetores de Suporte - SVM

O SVM é um classificador supervisionado, binário, que teve seus estudos iniciados por (VAPNIK; CHERVONENKIS, 1971) para análise de dados e reconhecimento de padrões, bastante usado na tarefa de classificação e regressão. Assumindo um conjunto de dados de entrada, tem como saída a predição de cada entrada, tendo por base duas possíveis classes.

A conceito que permeia o uso de SVM na tarefa de classificação, é a construção de um hiperplano que seja capaz de separar duas classes diferentes, com uma margem de separação ótima, que seria aquela que possuir a mesma distância para os elementos de ambas as classes.

Esses hiperplanos são obtidos por meio de subconjuntos de dados de treinamento - vetores de suporte, que definem qual é a melhor fronteira de decisão. Esses dados são mais propensos a serem separados de maneira linear em altas dimensões, Figura 4.2.

Figura 4.2 – Representação do SVM, (DOSCIATTI, 2015)



.

Inicialmente o SVM foi desenvolvido para o contexto de análise binária, entretanto algumas técnicas permitiram que o SVM fosse aplicado em problemas multiclasse. Para isso, combina-se os classificadores binários que foram gerados, em subproblemas de ordem

binária. Essa estratégia é conhecida como decomposição, conforme mostra o trabalho de (LORENA, 2006).

4.3.2 Naive Bayes - NB

O uso de métodos estatísticos tem sido amplamente aplicado em tarefas de aprendizado de máquina, como por exemplo, classificação textual. As duas principais vantagens do uso de aprendizado estatístico, principalmente os bayesianos está no fato de:

- poder acrescentar nas probabilidades calculadas o conhecimento de domínio que se tem;
- a capacidade das classificações feitas pelo algoritmo de aprendizado de máquina se basearem em evidências fornecidas, que podem aumentar ou diminuir as probabilidades das classes a serem observadas em uma nova instância que se quer classificar.

No entanto, as desvantagens do uso de aprendizado de máquina estatístico se dá, devido ao seu caráter estatístico, principalmente:

- muitas probabilidades devem ser calculadas;
- ocasionar um alto custo computacional. Uma das soluções para o custo do cálculo das probabilidades necessárias para o aprendizado do algoritmo é a aplicação do classificador Naive Bayes.

Naive Bayes é um método de aprendizado de máquina supervisionado, usado para classificação que considera as variáveis como independentes, por esse motivo é tido como ingênuo. É um bom método, simples de compreender e de fácil implementação, frequentemente aplicado em Processamento de Língua Natural. Esse método pode ser usado quando os atributos que descrevem as instâncias forem condicionalmente independentes dada a classificação.

O teorema de Bayes trata sobre probabilidade condicional. Isto é, qual a probabilidade de o evento A ocorrer, dado o evento B, o teorema de Bayes pode ser representado pela Equação 4.1, a qual pode ser interpretada da seguinte forma: probabilidade do evento A ocorrer dado o evento B é igual probabilidade do evento B ocorrer dado o evento A vezes a probabilidade do evento A sobre a probabilidade do evento B.

$$\mathrm{P}(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4.1}$$

Nesse contexto, supondo que A represente a classe e B, sejam as instâncias a serem classificadas, b_1, b_2 ... b_n, temos:

$$\mathrm{P}(classe|b_1...b_n) = \frac{P(b_1...b_n|classe)P(classe)}{P(b_1...b_n)} \tag{4.2}$$

Assim, para calcular a classe mais provável de uma nova instância, calcula-se a probabilidade de todas as possíveis classes e no final, escolhe-se a classe com a maior probabilidade como rótulo da nova instância. Isso estatisticamente equivale a maximizar a $\mathrm{P}(\mathrm{classe}|\mathrm{b}_1...b_n), sendoassim, deve-semaximarovalordonumeradorP(b_1...b_n|classe)P(classe)eassir$

$$argmaxP(classe|b_1...b_n) = argmaxP(b_1...b_n|classe)P(classe) \tag{4.3}$$

O Naive Bayes, supõe que todos os atributos b_2 ... b_n da instância que se quer classificar são independentes. Dessa forma, o cálculo o valor $P(b_1...b_n|classe)$, pode ser reduzido para $P(b_1|classe)x...xP(b_n|classe)$. Assim, a fórmula final pode ser expressa por:

$$argmaxP(classe|b_1...b_n) = argmax\Pi P(b_1|classe)P(classe) \tag{4.4}$$

Apesar de suas suposições aparentemente simplificadas, os classificadores ingênuos de Bayes têm funcionado muito bem em muitas situações do mundo real, notadamente na classificação de documentos. Eles exigem uma pequena quantidade de dados de treinamento para estimar os parâmetros necessários.

4.3.3 K-Nearest Neighbors - KNN

O método KNN é considerado um dos métodos de classificação mais antigos e simples (COVER; HART, 1967). Apesar da sua simplicidade, esse método tem alcançado bom desempenho em diferentes cenários, como em classificação textual.

O KNN é considerado um método preguiçoso, e um método preguiçoso simplesmente armazena os documentos de treino e realiza uma única etapa para classificar documentos.

Dado um documento de teste d, para classificá-lo o método KNN tradicionalmente realiza as seguintes atividades:

- a distância entre o documento d e cada um dos documentos de treino é calculada utilizando alguma medida de similaridade entre documentos, como a Euclidiana;
- Os k documentos de treino mais próximos, isto é, mais similares ao documento d são selecionados.

- O documento *d* é classificado em determinada categoria de acordo com algum critério de agrupamento das categorias dos k documentos de treino selecionados na etapa anterior.

O critério de similaridade adotado pelo KNN, exerce uma grande influência no desempenho do método. Esse critério é composto pela medida de similaridade, ou função de distância e pelo critério de seleção dos vizinhos. O critério de seleção determina a forma de escolha dos k vizinhos de um documento. Por exemplo, selecionar os k documentos de treino mais próximos do documento de teste d para um valor de k fixo é um critério de seleção tradicionalmente adotado pelo método kNN.

Um outro ponto importante que tem bastante influência no desempenho do método KNN é a definição do valor mais adequado para k. De um modo em geral, quando o conjunto de treino possui muitos elementos classificados incorretamente por um especialista, é preferível utilizar o método KNN com k = 1, caso contrário, com k > 1. Entretanto, para determinar o valor de k que o método KNN possui melhor desempenho é necessário a realização de experimentos (tentativa e erro) escolhendo diferentes valores para k.

De um modo geral, dado um exemplo de teste, o algoritmo KNN busca encontrar todos os possíveis exemplos do conjunto de treinamento que possuem semelhança com o dado de teste.

O KNN é considerado um método simples de categorização de texto (JIANG et al., 2012), no entanto é bastante sensível a ruídos nos dados de treinamento.

4.3.4 Medidas de Avaliação

Após costrução de um modelo de aprendizado de máquina, é necessário medir a qualidade dele de acordo com o que se espera obter, para isso faz-se uso de métricas, ou medidas de avaliação, que ajudam na avaliação da capacidade de acerto e erro do modelo gerado.

As métricas mostradas a seguir, são utilizadas em tarefas de classificação, e a maioria delas pode ser adaptada tanto para classificação binária quanto de múltiplas classes. Nas tarefas de classificação buscamos prever qual é a categoria a que uma amostra pertence.

- Acurácia: indica uma performance geral do modelo. Dentre todas as classificações, quantas o modelo classificou corretamente.

$$\text{Acurácia} = \frac{\text{Verdadeiro Positivo} + \text{Verdadeiro Negativo}}{\text{Total de amostras}} \tag{4.5}$$

- Precisão: dentre todas as classificações de classe Positivo que o modelo fez, quantas estão corretas.

$$\text{Precisão} = \frac{\text{Verdadeiro Positivo}}{\text{Verdadeiro Positivo + Falso Positivo}} \tag{4.6}$$

- Recall: dentre todas as situações de classe Positivo como valor esperado, quantas estão corretas.

$$\text{Recall} = \frac{\text{Verdadeiro Positivo}}{\text{Verdadeiro Positivo + Falso Negativo}} \tag{4.7}$$

- F1 - Score: média harmônica entre precisão e recall.

  Ela é muito boa quando você possui um dataset com classes desproporcionais, e o seu modelo não emite probabilidades. Isso não significa que não possa ser usada com modelos que emitem probabilidades, tudo depende do objetivo tarefa a ser realizada.

$$\text{F1 - Score} = \frac{2 * \text{precisão} * \text{recall}}{\text{precisão + recall}} \tag{4.8}$$

Quando a tarefa é um problema de classificação multiclasse, é importante observar as medidas de desempenho por classe e a classificação global que geralmente é avaliada por meio de média ponderada calculada em função do número de amostras de cada classe.

5

Experimentos e Resultados

Neste capítulo, serão apresentados os principais experimentos realizados com o corpus construído com os *tweets*, descrito no Capítulo 3 e outros encontrados na literatura, usando as técnicas de processamento de texto e algoritmos retratados ao longo deste trabalho. Ao final, será realizada uma análise geral dos resultados obtidos com esses algoritmos.

5.1 Aspectos Avaliados nos Experimentos

O estudo apresentado nesta pesquisa foi feito para análise do real sentimento dos usuários a respeito dos finalistas do MasterChef Profissionais, e para isso foi necessário abordar alguns pontos principais:

- O impacto da rotulação dos dados em diferentes números de classes;
- O impacto da representação do tipo *embedding* com relação à convencional, usando TF-IDF;
- O impacto da avaliação com as bases rotuladas com polaridade usando ambos os modelos de representação textual. Com intuito de entender como a variação do número de classes interfere nos resultados da classificação.

Para realizar as avaliações a respeito dos pontos levantados, foram utilizadas duas bases rotuladas com as emoções básicas (seis classes emocionais e o neutro):

- A base descrita no Capítulo 3;
- E a base "Corpus de Notícias"(DOSCIATTI; FERREIRA; PARAISO, 2015), com uma descrição da base mais a frente.

As bases originais possuem 7 classes de emoções, porém com o objetivo de avaliar o impacto no número de classes no processo de classificação, houve uma redução no número de classes avaliadas com base em um estudo feito por (JACK; GARROD; SCHYNS, 2014), que mostra que a incorporação de classes onde há um número significativo de sobreposição não sofreria alterações do ponto de vista psicológico, sendo assim, observa-se que em ambas as bases que há uma grande sobreposição entre os pares de classes "raiva"/"desgosto"e as classes "surpresa"/"medo". Logo, os dados da classe desgosto foram incorporados à classe raiva e os da classe surpresa foram incorporados à classe medo.

Com o objetivo de avaliar o impacto do número de classes a serem identificadas, as emoções usadas em ambas as bases foram reduzidas de sete para cinco classes. No entanto optou-se por manter o comparativo com as 7 emoções, afim de comparar os resultados e avaliar os benefícios que essa incorporação traria para os resultados.

Para a base do Capítulo 3 a distribuição dos dados após a redução das classes assumiu a seguinte configuração, conforme mostra a Tabela 5.1:

Tabela 5.1 – Distribuição das Classes de Emoções no Corpus após a incorporação

Emoção	**Total**
Alegria	855
Raiva	711
Medo	287
Tristeza	101
Neutro	596

Na seção 5.3 serão dados maiores detalhes sobre a base "Corpus de Notícias"e as distribuições dos dados.

Ambas as bases foram analisadas segundo a polaridade, com o intuito de analisar o desempenho dos modelos gerados, levando-se em consideração apenas a polaridade e as técnicas de representação textual.

Nas seções seguintes foram realizados os seguintes experimentos:

- 5.2 - Experimentos com a base apresentada no capítulo 3;
- 5.3 - Experimentos com a base "Corpus de Notícias";
- 5.4 - Experimentos com base rotulada apenas com polaridade.

Para validar os modelos de cada experimento, foi usada a técnica de validação cruzada (WITTEN et al., 2016). Esse procedimento consiste em dividir a base de dados em k partes, essas partes são chamadas de *folds*. Uma dessas partes é escolhida para testar o modelo, enquanto o restante é utilizado na fase de treinamento, isso é feito repetidamente

até que o modelo seja treinado e testado com todas as partes. Em todos os testes, foram utilizados 10-*folds*. Esse processo avalia a capacidade de generalização de um modelo a partir de um conjunto de dados. Devido a essa generalização, os problemas de variância nos dados são minimizados.

5.2 Experimentos com a Base Apresentada no Capítulo 3

A Tabela 5.2 mostra os resultados do processamento dos dados com o algoritmo SVM para dados emocionais.

Tabela 5.2 – Execução SVM - Emoções

SVM - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	50.5549%	**55.647%**	52.980%	**58.863%**
Precisão	0.504	0.558	0.511	0.589
Recall	0.505	0.556	0.515	0.537
F1 - Score	0.504	0.557	0.513	0.562

Na Tabela 5.3 mostra os resultados do processamento dos dados com o algoritmo SVM para dados com polaridade.

Tabela 5.3 – Execução SVM - Polaridade

SVM - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	62.196%	**68.667%**
Precisão	0.627	0.697
Recall	0.622	0.691
F1 - Score	0.624	0.694

Para os experimentos realizados com o SVM, foram usadas as configurações padrão, com *kernel* RBF.

Nos experimentos realizados com o algoritmo SVM, os resultados com as cinco classes obtiveram os melhores resultados, com destaque para aqueles que fizeram uso do Word2Vec.

O mesmo pode ser observado com os resultadas com a mesma base rotulada com a polaridade, sendo que o dados representados com Word2Vec apresentaram melhores resultados.

Na Tabela 5.4 o conjunto de dados foram submetidos ao classificador NB, para os dados emocionais.

Tabela 5.4 – Execução Naive Bayes - Emoções

NB - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	42.902%	**47.843%**	49.922%	**53.216%**
Precisão	0.417	0.455	0.481	0.498
Recall	0.429	0.478	0.463	0.501
F1 - Score	0.423	0.466	0.472	0.499

Já na Tabela 5.5 o conjunto de dados foram submetidos ao classificador NB, para os dados com polaridade.

Tabela 5.5 – Execução Naive Bayes - Polaridade

NB- Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	56.588%	**61.922%**
Precisão	0.564	0.615
Recall	0.566	0.621
F1 - Score	0.565	0.618

Com o uso do NB para classificação, os experimentos que obtiveram os melhoras resultados para emoção foram os com cinco classes e principalmente o que usou o Word2Vec para representação textual, o mesmo pode ser observado para análise da polaridade, o Word2Vec apresentou um resultado mais significativo se comparado com o TF-IDF.

A seguir serão mostrados os experimentos usando o algoritmo kNN, as escolhas para os valores de k, basicamente foram com base em experimentos com base em diferentes valores para k, no entanto para valores maiores de k o desempenho caia bastante em relação aos valores mostrados nas tabelas. Um fator que contribui para isso é o fato de que possa haver ruídos no conjunto de teste, nesse caso, pode haver muitos elementos classificados incorretamente pelos anotadores, nesse caso é preferível usar o método KNN com $k = 1$, e $k = 5$ pois valores maiores que esse, o desempenho apresentava uma queda maior.

Nos experimentos das Tabelas 5.6 e 5.7, mostram os resultados para o classificador kNN, que foram usados os valores de $k = 1$.

Tabela 5.6 – Execução kNN1 - Emoções

kNN1 - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	40.471%	**44.980%**	44.196%	**51.20%**
Precisão	0.4	0.462	0.421	0.481
Recall	0.405	0.45	0.475	0.499
F1 - Score	0.402	0.456	0.446	0.490

Tabela 5.7 – Execução kNN1 - Polaridade

kNN1 - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	52.275%	**60.824%**
Precisão	0.561	0.571
Recall	0.523	0.58
F1 - Score	0.541	0.575

As Tabelas 5.8 e 5.9 mostram os resultados para $k = 5$.

Tabela 5.8 – Execução kNN5 - Emoções

kNN5 - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	**41.373%**	36.392%	**50.627%**	45.451%
Precisão	0.429	0.421	0.522	0.446
Recall	0.414	0.364	0.579	0.542
F1 - Score	0.421	0.387	0.549	0.449

Tabela 5.9 – Execução kNN5 - Polaridade

kNN5 - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	41.490%	**50.863%**
Precisão	0.535	0.596
Recall	0.415	0.517
F1 - Score	0.467	0.554

Nos experimentos com kNN5, os resultados já assumiram valores diferentes dos demais algoritmos, para os dados emocionais, as configurações com sete classes e com uso de TF-IDF apresentaram os melhores resultados. Já para os dados de polaridade o uso do Word2Vec apresenta o melhor resultado.

A diferença entre os experimentos com kNN para k =1 e 5 pode ser pela presença de ruídos na base, pois o kNN é um algoritmo bastante sensível a dados ruidosos principalmente para valores mais elevados de k.

Esses são os experimentos realizados com a base construída para o trabalho, a ideia geral era submeter esse corpus a uma análise de algoritmos de aprendizado de máquina, para identificação do desempenho da base na busca pela identificação do real sentimento dos usuários, usando diferentes técnicas de classificação e formas de vetorização dos dados textuais.

Observando os resultados das tabelas, percebe-se que o classificador SVM apresenta o melhor desempenho entre os algoritmos usados neste trabalho, tanto para a tarefa de classificação das emoções quanto a de polaridade. Umas das principais justificativas é o fato do SVM tratar dados multiclasse, por meio da estratégia de combinar os classificadores gerados em subproblemas binários, a qual é chamada de decomposição. Isso permite criar hiperplanos melhores para a tarefa de classificação. E ainda o uso do Word2Vec obteve melhor desempenho em relação a técnica de vetorização TF-IDF.

Em relação as tarefas de classificação das emoções e polaridade, em todos os cenários a polaridade apresentou melhor desempenho, usando o Word2Vec, deixando evidente que o uso de técnicas com uma maior respresentação dos dados para extração de **features** melhora de maneira significativa os modelos computacionais de inferência textual.

Apesar de todos os esforços de aplicações de técnicas de PLN, o processo de inferência multiclasse para Análise de Sentimento ainda envolvem fatores que dificultam essa análise, como por exemplo a subjetividade dos dados, da anotação, uso de expressões da internet no texto, ironia, entre outros. Já na análise segundo a polaridade estas questões existem, porém a quantidade de classe são menores e elas possuem delimitações bem definidas umas das outras, situação que não é tão simples quando trabalha-se com 6 classes, onde essas classes muitas vezes se sobrepõem, prejudicando o processo de anotação e avaliação dos dados.

A seguir, os experimentos mostrados são de outras bases de dados, para análise de sentimentos.

5.2.1 Análise de Sentimento - Outras bases

5.2.1.1 Corpus de Notícias

Este é um corpus costruído em (DOSCIATTI; FERREIRA; PARAISO, 2015) a partir de dados de notícias jornalísticas para Análise de Sentimento para o Português brasileiro, rotulados de acordo com as seis emoções básicas.

Essa base, possui as seguintes distribuições de dados: 542 neutro, 455 para tristeza, 262 sendo desgosto, 252 para surpresa, 222 para medo, 184 para alegria e 83 como raiva.

No trabalho original, os dados foram submetidos a uma análise com o algoritmo SVM, e com vetorização TF-IDF dos dados, conforme mostra a Tabela 5.10. Para com-

plementar os experimentos, foram acrescentados experimentos usando Word2Vec com processamento pelos algoritmos SVM, Naive Bayes e KNN ($k = 1$ e $k = 5$), no total são 2000 textos. Os experimentos foram realizados usando as emoções básicas mais a neutra, totalizando 7 classes, e além dos experimentos com essas classes, realizamos a junção das classes, onde raiva/desgosto passaram a ser apenas raiva e surpresa/medo passaram a ser apenas medo, conforme mostrado na secção 5.1, com a finalidade de avaliar os dados com classes reduzidas, para a verificar se a redução de classes contribui para um melhor desempenho do modelo gerado.

Tabela 5.10 – Execução SVM - Emoções - Corpus de Notícias

SVM - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	60.30%	**64.60%**	63.70%	**69.90%**
Precisão	0.657	0.690	0.692	0.668
Recall	0.541	0.636	0.633	0.612
F1 - Score	0.593	0.663	0.660	0.639

A Tabela 5.11 mostra os resultados para polaridade, essa manipulação dos dados de conteúdo emocional para polaridade, foi com base no trabalho da autora (DOSCIATTI, 2015), que relata essa transformação dos dados. Essa transformação seguiu a lógica onde os dados rotulados como:

- Alegria passaram assumir o rótulo positivo;
- Tristeza, desgosto, medo e raiva passam assumir o rótulo negativo;
- Neutro continua sendo neutro;
- Surpresa foram descartados pois nessa categoria podem haver interpretações tanto positivas quanto negativas.

Nesse contexto a base passou a ter 1.748 dados, distribuídos da seguinte forma: 184 dados positivos, 1022 dados negativos e 542 dados neutros.

Tabela 5.11 – Execução SVM - Polaridade - Corpus de Notícias

SVM - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	67.83%	**72.38%**
Precisão	0.601	0.721
Recall	0.597	0.699
F1 - Score	0.599	0.710

Nos resultados apresentados, o uso de 5 classes e aplicação dos Word2Vec continuam apresentando os melhores resultados. O mesmo acontece para dados rotulados com polaridade.

Na Tabela 5.12 são apresentados os resultados dos dados submetidos ao classificador Naive Bayes.

Tabela 5.12 – Execução NB - Emoções - Corpus de Notícias

NB - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	48.94%	**53.21%**	52.33%	**59.71%**
Precisão	0.501	0.528	0.521	0.536
Recall	0.489	0.532	0.513	0.541
F1 - Score	0.495	0.530	0.517	0.538

A Tabela 5.13 mostra os resultados com dados segundo a polaridade.

Tabela 5.13 – Execução Naive Bayes - Polaridade - Corpus de Notícias

NB - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	70.36%	**76.29%**
Precisão	0.701	0.792
Recall	0.704	0.783
F1 - Score	0.702	0.787

Com o uso do NB para a base emocional e de polaridade o uso do Word2Vec melhorou o desempenho do algoritmo.

Nas Tabelas 5.14 e 5.15, são apresentados os resultados dos experimentos com o classificador kNN, para valores de $k = 1$, com as bases rotuladas com emoções e polaridade, respectivamente.

Tabela 5.14 – Execução kNN1 - Emoções - Corpus de Notícias

kNN1 - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	29.22%	**35.61%**	32.41%	41.30%
Precisão	0.341	0.401	0.471	0.502
Recall	0.292	0.356	0.432	0.479
F1 - Score	0.315	0.377	0.451	0.490

As Tabelas 5.16 e 5.17 mostram os resultados para $k = 5$.

Tabela 5.15 – Execução kNN1 - Polaridade - Corpus de Notícias

kNN1 - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	47.69%	**53.88%**
Precisão	0.592	0.603
Recall	0.477	0.622
F1 - Score	0.528	0.612

Tabela 5.16 – Execução kNN5 - Emoções - Corpus de Notícias

kNN5 - Emoções				
	TF - IDF		Word2Vec	
Parâmetro	**7 Classes**	**5 Classes**	**7 Classes**	**5 Classes**
Acurácia	27.91%	**37.92%**	30.39%	**43.01%**
Precisão	0.392	0.389	0.459	0.463
Recall	0.279	0.379	0.421	0.452
F1 - Score	0.326	0.384	0.439	0.457

Tabela 5.17 – Execução kNN5 - Polaridade - Corpus de Notícias

kNN5 - Polaridade		
Parâmetro	TF - IDF	Word2Vec
Acurácia	46.43%	**50.13%**
Precisão	0.640	0.569
Recall	0.464	0.553
F1 - Score	0.538	0.561

Com o algoritmo kNN os resultados para as cinco classes e com o uso do Word2Vec apresentaram os melhores resultados, tanto para dados emocionais e polaridade.

Apesar de ambas as bases possuírem domínios diferentes, pode-se inferir algumas questões importantes dos experimentos. Nas duas bases, o uso do Word2Vec como técnica de vetorização, aumentou bastante o desempenho dos classificadores. Isso se deve ao fato de que essa técnica guarda informações semânticas das palavras, que aplicadas no contexto de Análise de Sentimento proporcionam um ganho significativo quando comparado a técnica TF-IDF que apenas guarda a frequência dos termos em uma sentença, sem carregar valor semântico para análise. E essa análise semântica permite uma melhor inferência do classificador em relação às emoções.

Questões como o domínio também exercem grande influência na análise, pois dependendo do domínio usado, o processo de anotação pode ser fortemente influenciado. Em dados jornalísticos por exemplo, os dados a serem analisados possuem um caráter mais objetivo e impessoal, o que acaba limitando algumas possibilidades de interpretação, enquanto que em textos extraídos de redes sociais, onde o seu conteúdo é gerado por usuários que depositam grande teor emocional nas postagens, dando uma maior possibilidade de entendimento e interpretação por parte do leitor (anotador) na hora de realizar

a anotação.

5.2.1.2 Experimentos com Base Rotulada apenas com Polaridade

Além do uso de bases com classes de emoções, foram realizados alguns experimentos com bases originalmente rotuladas apenas com polaridade, como é o caso da base "Tweets para Análise de Sentimentos em Português (TAS-PT)", que não possui domínio específico, e foi construída para realizar análises de sentimentos de *tweets*, capturados no mês de maio de 2017.

Os dados foram coletados usando a API do *Twitter* e rotulados automaticamente (Tweets com emoticons ':)' ou ':-)' foram rotulados como positivos e tweets com emoticons ':(' ou ':-(' foram rotulados como negativos) de acordo com a sua polaridade (positiva ou negativa). Cada uma das classes possuem 38119 textos anotados. O tratamento dos dados, foram os mesmos aplicados nas bases dos experimentos anteriores.

A Tabela 5.18 mostra a execução do SVM para a base TAS - PT.

Tabela 5.18 – Execução SVM - Polaridade - TAS - PT

SVM - Polaridade		
Parâmetro	TF IDF	Word2Vec
Acurácia	61.7%	**69.6%**
Precisão	0.581	0.682
Recall	0.694	0.701
F1 - Score	0.697	0.725

A Tabela 5.19 mostra a execução do NB para a base TAS - PT.

Tabela 5.19 – Execução NB - Polaridade - TAS - PT

NB - Polaridade		
Parâmetro	TF IDF	Word2Vec
Acurácia	57.9%	**62.8%**
Precisão	0.581	0.682
Recall	0.537	0.658
F1 - Score	0.558	0.670

A Tabela 5.20 mostra a execução do kNN com $k = 1$ para a base TAS - PT.

Tabela 5.20 – Execução kNN1 - Polaridade - TAS - PT

kNN1 - Polaridade		
Parâmetro	TF IDF	Word2Vec
Acurácia	51.1%	**55.5%**
Precisão	0.513	0.524
Recall	0.497	0.513
F1 - Score	0.505	0.518

A Tabela 5.21 mostra a execução do kNN com $k = 5$ para a base TAS - PT.

Tabela 5.21 – Execução kNN5 - Polaridade - TAS - PT

kNN5 - Polaridade		
Parâmetro	TF IDF	Word2Vec
Acurácia	46.1%	**51.3%**
Precisão	0.502	0.593
Recall	0.481	0.568
F1 - Score	0.491	0.580

Com base nos resultados, percebe-se que o uso do Word2Vec, melhora o desempenho dos classificadores se comparado ao TF - IDF.

Observando todos os experimentos apresentados nesta seção, podemos inferir que o uso de uma técnica de vetorização textual, como o Word2Vec, capaz de extrair dos dados valores que possam dar uma maior representatividade semântica às palavras constituintes do vetor, permitindo assim, uma melhor classificação do real sentimento de textos independente se a base for rotulada com classes emocionais ou com polaridade.

5.3 Classificação de Novos Exemplos

Uma vez realizados testes para validar o corpus e ter construído os modelos necessários para realizar a classificação de novos exemplos, os experimentos a seguir tem por objetivo analisar a classificação de novos exemplos passados ao modelo e poder realizar inferências a respeito do sentimento dos usuários em relação aos três finalistas do programa.

Para isso foi utilizado um conjunto de dados, com 450 textos anotados de acordo com as emoções básicas e com a polaridade de cada texto. Esse conjunto de dados visa avaliar o quão correto é a classificação dos modelos, tendo em vista que podemos comparar a saída dada pelo modelo, com a saída real, ou seja, a saída gerada pelos anotadores. Os dados foram anotados seguindo os mesmos critérios e passos usados na construção do corpus.

A Tabela 5.22 mostra os valores reais e os classificados pelos modelos para o algoritmo SVM, onde temos os dados reais rotulados com as emoções e polaridade.

Tabela 5.22 – Classificação de Novos Exemplos - SVM

SVM - 7 Emoções				
Dados			TF-IDF	Word2Vec
Classe	Real	Porcentagem	% Classificados	% Classificados
Alegria	1782	40%	38%	42%
Tristeza	18	4%	1%	5%
Raiva	38	8%	6%	4%
Surpresa	27	6%	4%	4%
Medo	20	4%	2%	9%
Desgosto	71	16%	12%	12%
Neutro	98	22%	37%	24%
SVM - 5 Classes				
Alegria	178	40%	37%	39%
Tristeza	18	4%	1%	3%
Medo	47	10%	6%	7%
Raiva	109	24%	23%	21%
Neutro	98	22%	33%	30%
SVM - Polaridade				
Positivo	170	38%	37%	34%
Neutro	188	42%	30%	39%
Negativo	92	20%	33%	27%

A Tabela 5.23 mostra os dados da classificação do modelo para dados usando o algoritmo Naive Bayes, para classes emocionais e de polaridade. Percebe-se que o uso da técnica Word2Vec apresenta um desempenho melhor em relação ao TF-IDF.

Tabela 5.23 – Classificação de Novos Exemplos - NB

NB - 7 Emoções				
Dados			TF-IDF	Word2Vec
Classe	Real	Porcentagem	% Classificados	% Classificados
Alegria	1782	40%	44%	35%
Tristeza	18	4%	2%	6%
Raiva	38	8%	10%	9%
Surpresa	27	6%	5%	8%
Medo	20	4%	6%	5%
Desgosto	71	16%	16%	14%
Neutro	98	22%	16%	23%
NB - 5 Classes				
Alegria	178	40%	46%	42%
Tristeza	18	4%	3%	5%
Medo	47	10%	7%	9%
Raiva	109	24%	35%	30%
Neutro	98	22%	9%	14%
NB - Polaridade				
Positivo	170	38%	56%	42%
Neutro	188	42%	1%	15%
Negativo	92	20%	43%	43%

A Tabela 5.24 mostra a classificação do modelo para novos dados, usando o algoritmo kNN com valor de $k = 1$ para classes emocionais e de polaridade.

Tabela 5.24 – Classificação de Novos Exemplos - kNN1

kNN1 - 7 Emoções				
Dados			TF-IDF	Word2Vec
Classe	Real	Porcentagem	% Classificados	% Classificados
Alegria	1782	40%	29%	37%
Tristeza	18	4%	1%	6%
Raiva	38	8%	7%	5%
Surpresa	27	6%	5%	3%
Medo	20	4%	1%	7%
Desgosto	71	16%	14%	15%
Neutro	98	22%	43%	27%
kNN1 - 5 Classes				
Alegria	178	40%	29%	35%
Tristeza	18	4%	1%	2%
Medo	47	10%	7%	12%
Raiva	109	24%	21%	21%
Neutro	98	22%	42%	30%
kNN1 - Polaridade				
Positivo	170	38%	31%	35%
Neutro	188	42%	40%	38%
Negativo	92	20%	29%	27%

A Tabela 5.25 mostra a classificação do modelo para novos dados, usando o algoritmo kNN com valor de $k = 5$.

Tabela 5.25 – Classificação de Novos Exemplos - kNN5

kNN5 - 7 Emoções				
Dados			TF-IDF	Word2Vec
Classe	Real	Porcentagem	% Classificados	% Classificados
Alegria	1782	40%	15%	33%
Tristeza	18	4%	0%	2%
Raiva	38	8%	4%	6%
Surpresa	27	6%	1%	4%
Medo	20	4%	0%	3%
Desgosto	71	16%	11%	13%
Neutro	98	22%	69%	39%
KNN5 - 5 Emoções				
Alegria	178	40%	14%	29%
Tristeza	18	4%	0%	1%
Medo	47	10%	2%	8%
Raiva	109	24%	17%	13%
Neutro	98	22%	67%	49%
KNN5 - Polaridade				
Positivo	170	38%	15%	27%
Neutro	188	42%	67%	51%
Negativo	92	20%	18%	22%

Os experimentos exibidos nas tabelas acima, mostram os resultados dos modelos para novos dados, com a finalidade de identificar o percentual de acerto dos modelos e ter uma visão de como os modelos classificam novos dados e assim ter uma maior confiabilidade nas saídas de dados em que os rótulos são desconhecidos. De um modo geral, os modelos conseguem ter um percentual de acerto aceitável, mas podemos observar que os dados vetorizados com o Word2Vec apresentam um percentual de classificação mais próximo dos dados reais, mostrando assim que o uso dessa técnica possibilita uma melhor classificação para a tarefa de Análise de Sentimento, por levar em consideração aspectos semânticos das palavras, que é mais indicado para essa tarefa.

5.3.1 Inferência das Emoções do Usuários

Após esse processo, o passo seguinte foi submeter dados novos, sem rótulos, para serem classificados usando os modelos, com a finalidade de compreender os sentimentos dos usuários em relação aos participantes do programa.

Para isso, dos *tweets* capturados ao longo de todos os episódios do programa, alguns foram selecionados para construção do corpus e os demais, que continham referência aos

finalistas foram selecionados, separados em três arquivos, cada um representando um participante. Assim os arquivos possuem as seguintes configurações:

- Arquivo participante Irina: total de 25103 *tweets*.
- Arquivo participante Pablo: total de 23234 *tweets*.
- Arquivo participante Francisco: total de 33091 *tweets*.

Esses *tweets* foram selecionados usando uma *string* de busca com o nome de cada um dos participantes, com a finalidade de submeter ao processo de classificação. Os textos de cada arquivo passaram pelo mesmo processo de pré-processamento que os textos do corpus, passando por cada uma das etapas de tratamento textual.

Após esse processamento, é possível inferir de um modo geral, o percentual de cada emoção que os usuários têm em relação a cada finalista. A confiança desses percentuais de classificação podem ser garantidos com base nos experimentos da seção anterior e nas taxas de classificação inferidas e comparadas com as taxas reais de cada classe dos dados rotulados usados nos experimentos.

Na Tabela 5.26 estão os resultados da inferência do modelo executado pelo algoritmo SVM, paras as classes emocionais e de polaridade.

Tabela 5.26 – Inferência de dados com modelo SVM

SVM - 7 Emoções						
Emoção	Irina		Pablo		Francisco	
	TF-IDF	Word2Vec	TF-IDF	Word2Vec	TF-IDF	Word2Vec
Alegria	43.1%	48.8%	28.2%	32.3%	16.3%	12.7%
Desgosto	5.7%	6.3%	12.2%	10.7%	29.1%	34.0%
Medo	1.6%	1.4%	0.4%	0.2%	0.4%	2.1%
Neutro	44.2%	39.3%	51.3%	50.9%	43.0%	37.7%
Raiva	2.5%	2.0%	3.5%	2.9%	7.0%	10.2%
Surpresa	1.6%	1.0%	3.8%	2.7%	3.3%	2.5%
Tristeza	1.3%	1.2%	0.6%	0.4%	1.0%	0.8%
SVM - 5 Emoções						
Alegria	47.4%	51.3%	27.0%	30.6%	13.5%	11.1%
Tristeza	1.3%	2.7%	0.6%	1.7%	0.9%	1.1%
Raiva	9.6%	7.5%	19.3%	21.4%	43.6%	50.3%
Medo	2.9%	3.5%	4.1%	5.5%	4.0%	4.8%
Neutro	38.8%	35.1%	48.9%	40.8%	38.0%	32.7%
SVM - Polaridade						
Negativo	14.9%	16.0%	28.1%	31.2%	57.6%	63.5%
Positivo	52.3%	55.1%	27.9%	28.4%	12.4%	13.9%
Neutro	32.8%	28.9%	44.0%	40.4%	30.0%	22.6%

Na Tabela 5.27 estão os resultados da inferência do modelo executado pelo algoritmo NB, paras as classes emocionais e de polaridade.

Tabela 5.27 – Inferência de dados com modelo NB

NB - 7 Emoções						
Emoção	Irina		Pablo		Francisco	
	TF-IDF	Word2Vec	TF-IDF	Word2Vec	TF-IDF	Word2Vec
Alegria	80.6%	77.9%	46.4%	43.7%	14.0%	12.0%
Desgosto	3.7%	4.9%	9.3%	8.9%	26.9%	30.5%
Medo	5.5%	3.8%	2.8%	2.9%	3.0%	4.1%
Neutro	2.4%	3.1%	23.9%	28.1%	32.8%	30.1%
Raiva	1.0%	3.3%	6.5%	5.6%	10.7%	12.5%
Surpresa	3.2%	2.8%	9.3%	8.3%	9.6%	8.0%
Tristeza	3.5%	4.2%	1.8%	2.5%	2.9%	2.8%
NB - 5 Emoções						
Alegria	83.1%	84.4%	47.9%	45.1%	13.5%	11.1%
Tristeza	3.6%	2.3%	2.0%	3.9%	2.6%	3.9%
Raiva	5.2%	6.7%	23.3%	19.7%	73.1%	75.1%
Medo	5.9%	3.4%	7.4%	8.9%	9.3%	7.1%
Neutro	2.2%	3.2%	19.5%	22.4%	1.6%	2.8%
NB - Polaridade						
Negativ	14.1%	12.7%	30.7%	32.8%	85.8%	89.1%
Positivo	84.6%	85.2%	68.1%	64.3%	13.8%	11.4%
Neutro	1.3%	2.1%	1.2%	2.9%	0.4%	0.5%

Na Tabela 5.28 estão os resultados da inferência do modelo executado pelo algoritmo KNN com $k = 1$, paras as classes emocionais e de polaridade.

Tabela 5.28 – Inferência de dados com modelo kNN1

kNN1 - 7 Emoções						
Emoção	Irina		Pablo		Francisco	
	TF-IDF	Word2Vec	TF-IDF	Word2Vec	TF-IDF	Word2Vec
Alegria	27.8%	35.6%	27.3%	25.1%	17.1%	13.8%
Desgosto	1.4%	2.7%	8.9%	6.4%	19.7%	23.6%
Medo	2.4%	1.9%	0.9%	1.7%	0.6%	0.9%
Neutro	58.6%	51.1%	49.3%	53.7%	48.7%	38.5%
Raiva	2.3%	3.8%	5.0%	4.1%	9.2%	21.3%
Surpresa	2.8%	1.7%	6.5%	7.4%	2.9%	1.1%
Tristeza	4.6%	3.2%	2.1%	1.6%	1.8%	0.8%
kNN1 - 5 Emoções						
Alegria	27.7%	51.9%	27.1%	30.2%	16.7%	12.1%
Tristeza	4.3%	1.1%	2.2%	1.3%	1.7%	4.7%
Raiva	4.3%	2.5%	13.2%	11.4%	30.6%	42.8%
Medo	5.5%	3.9%	8.1%	10.9%	3.5%	6.6%
Neutro	58.2%	40.6%	49.5%	46.2%	47.4%	33.8%
kNN1 - Polaridade						
Negativo	14.8%	12.7%	20.7%	18.9%	39.8%	51.3%
Positivo	32.1%	35.1%	31.1%	29.7%	13.6%	14.5%
Neutro	53.1%	52.2%	48.2%	51.4%	46.6%	34.2%

Na Tabela 5.29 estão os resultados da inferência do modelo executado pelo algoritmo KNN com $k = 5$, paras as classes emocionais e de polaridade.

Tabela 5.29 – Inferência de dados com modelo kNN5

kNN5 - 7 Emoções						
Emoção	Irina		Pablo		Francisco	
	TF-IDF	Word2Vec	TF-IDF	Word2Vec	TF-IDF	Word2Vec
Alegria	10.2%	14.6%	15.8%	17.3%	7.0%	4.9%
Desgosto	0.6%	31.0%	5.7%	4.1%	20.1%	29.7%
Medo	0.0%	0.8%	0.0%	1.3%	0.0%	1.2%
Neutro	87.3%	48.0%	75.6%	63.2%	70.6%	38.1%
Raiva	0.3%	1.4%	1.0%	3.1%	2.1%	21.9%
Surpresa	1.3%	2.1%	1.7%	8.8%	0.2%	2.3%
Tristeza	0.2%	2.1%	0.2%	2.2%	0.1%	1.9%
kNN5 - 5 Emoções						
Alegria	9.9%	17.6%	13.3%	15.9%	5.0%	3.6%
Tristeza	0.2%	2.2%	0.3%	1.7%	0.0%	0.7%
Raiva	1.4%	2.7%	8.4%	8.3%	24.5%	27.9%
Medo	1.5%	3.4%	2.4%	1.4%	0.2%	2.4%
Neutro	87.1%	74.1%	75.7%	72.7%	70.3%	65.4%
kNN5 - Polaridade						
Negativo	3.1%	4.4%	11.5%	12.9%	26.4%	30.1%
Positivo	11.2%	14.7%	13.0%	15.2%	3.1%	4.7%
Neutro	85.8%	80.9%	75.5%	71.9%	70.5%	65.2%

Os dados expostos nas tabelas 5.27, 5.28 e 5.29, tem por finalidade mostrar o sentimento dos usuários em relação a cada finalista do programa. Eles mostram o percentual de cada emoção sobre os finalistas. É importante ressaltar que essa análise é feita com dados coletados ao longo de todo o programa, então, essa análise reflete a trajetória completa do participante.

O objetivo desses experimentos foi de identificar o real sentimento em uma base de conhecimento, para isso fez-se uso de algoritmos clássicos, aplicando técnicas de vetorização textual (Word2Vec), que tem por objetivo, melhorar a representação das *features* a serem analisadas, uma vez que refletem as interações entre as palavras diante de um contexto específico, onde um dado termo possa ter um significado diferente se colocado em outra sentença por exemplo.

O uso de técnicas mais tradicionais de vetorização (TF-IDF) foi com o objetivo de estabelecer um comparativo com técnicas mais tradicionais de representação textual e estabelecer uma análise comparativa entre o tradicional e técnicas mais recentes, que buscam melhorias nas representações das palavras, atribuindo a tarefa de Análise de Sentimento um carácter mais próximo do real, daquilo que o usuário deseja transmitir em suas postagens.

6

Conclusão e Trabalhos Futuros

6.1 Conclusão

O trabalho descrito neste documento, tem por objetivo estudar aspectos da Análise de Sentimentos com relação ao número de classes de emoções e representação dos textos. Para a partir de então, identificar o real sentimento dos usuários a respeito dos finalistas do MasterChefe Profissionais, a partir das seis emoções básicas propostas por (EKMAN; FRIESEN, 1978), (alegria, raiva, tristeza, desgosto, medo e surpresa) e mais a classe neutra, em *tweets*, por meio dos classificadores SVM, Naive Bayes e kNN.

Para a realização do trabalho, foi necessário a construção da base de conhecimento-corpus, para posterior inferência por parte dos classificadores. Nesse processo de construção do corpus até o seu uso efetivo com os classificadores, diversos processos de tratamento do texto foram usados, entre eles o uso de técnicas de reapresentação textual, tais como TF-IDF e Word2Vec, que ao longo dos experimentos acabaram tendo importância significativa para a análise, tendo em vista a diferença de resultado entres as duas técnicas.

A análise feita com dados baseados em emoções e os de polaridade, mostrou que para em todas as situações o classificador apresentou melhor de desempenho na tarefa, com valor de acurácia de 58.86% para classificação emocional e 68.66% para polaridade. Isso devido ao fato desse classificar se adequar para problemas multiclasse.

Outras bases também foram utilizadas, com o objetivo de avaliar o uso do Word2Vec, evidenciando que essa técnica de representação tem sido bastante promissora principalmente em tarefas de Análise de Sentimentos e classificação de polaridade.

Mas ao final de todo o processo, foi possível realizar a inferência do sentimento dos usuários, mostrados na seção 5.2.1, em que para cada um dos classificadores, há uma inferência sobre as emoções expressas pelos usuários. Não é possível realizar uma comparação dos resultados, porém, pode-se confiar nos resultados com base nos experimentos onde dados novos, porém com rótulos conhecidos foram classificados pelos modelos, e assim pôde-se estabelecer um comparativo entre as porcentagens reais de classes e a porcenta-

gem de classes inferidas pelos modelos. O que permite ter uma maior confiabilidade na saída do classificador.

Os resultados obtidos neste trabalho, se comparados a outras técnicas de classificação, com outros domínios, que os resultados não são promissores. Mas quando falamos da tarefa de Análise de Sentimento, principalmente com classes emocionais, são resultados significativos. Uma vez que os desafios desse tipo de classificação ainda são grandes para a Área de Aprendizado de Máquina e PLN, pois os dados analisados são provenientes de usuários reais, que expressam opiniões baseadas em momentos específicos, com uso informal da escrita, em sua maioria.

A pesquisa desenvolvida neste trabalho resultou em contribuições científicas:

- DOS SANTOS, Allisfrank; JÚNIOR, Jorge Daniel Barros; DE ARRUDA CAMARGO, Heloisa. Annotation of a Corpus of Tweets for Sentiment Analysis. In: International Conference on Computational Processing of the Portuguese Language. Springer, Cham, 2018. p. 294-302.

6.2 Trabalhos Futuros

Como trabalhos futuros, pretende-se extender o estudo desta pesquisa, para análise textual para o contexto de Flúxo Contínuo de Dados - FCD, tendo em vista que os dados extraídos do *Twitter*, tem capacidade de geração de dados textual potencialmente infinitos, permitindo assim, com bases nos estudos de classificadores para FCD, realizar essa avaliação dos dados em tempo real.

Para isso, algoritmos como o *Very Fast Decison Tree - VFDT* e *Vertical Hoeffding Tree - VHT*, foram estudados para avaliar suas aplicações em dados textuais.

Estudos preliminares foram realizados, conforme descritos no Apêndice ao final deste trabalho.

Referências

AHMAD, M. et al. Svm optimization for sentiment analysis. *International Journal of Advanced Computer Science and Applications*, v. 9, 04 2018.

ALM, C. O.; ROTH, D.; SPROAT, R. Emotions from text: machine learning for text-based emotion prediction. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the conference on human language technology and empirical methods in natural language processing*. [S.l.], 2005. p. 579–586.

AMAN, S.; SZPAKOWICZ, S. Identifying expressions of emotion in text. In: SPRINGER. *International Conference on Text, Speech and Dialogue*. [S.l.], 2007. p. 196–205.

ARAÚJO, M.; GONÇALVES, P.; BENEVENUTO, F. Measuring sentiments in online social networks. In: ACM. *Proceedings of the 19th Brazilian symposium on Multimedia and the web*. [S.l.], 2013. p. 97–104.

ARNOLD, M. B. Emotion and personality. vol. i. psychological aspects. Columbia Univer. Press, 1960.

ARTSTEIN, R.; POESIO, M. Bias decreases in proportion to the number of annotators. In: *Proceedings of FG-MoL 2005: The 10th conference on Formal Grammar and The 9th Meeting on*. [S.l.: s.n.], 2009. v. 139.

BEN-HAIM, Y.; TOM-TOV, E. A streaming parallel decision tree algorithm. *Journal of Machine Learning Research*, v. 11, n. Feb, p. 849–872, 2010.

BENABDALLAH, A.; ABDERRAHIM, M. A.; ABDERRAHIM, M. E.-A. Extraction of terms and semantic relationships from arabic texts for automatic construction of an ontology. *International Journal of Speech Technology*, Springer, v. 20, n. 2, p. 289–296, 2017.

BENGIO, Y. et al. A neural probabilistic language model. *Journal of machine learning research*, v. 3, n. Feb, p. 1137–1155, 2003.

BERTAGLIA, T. F. C.; NUNES, M. d. G. V. Exploring word embeddings for unsupervised textual user-generated content normalization. In: *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*. [S.l.: s.n.], 2016. p. 112–120.

BIFET, A.; FRANK, E. Sentiment knowledge discovery in twitter streaming data. In: SPRINGER. *International conference on discovery science*. [S.l.], 2010. p. 1–15.

BRITO, E. *Mineração de Textos: Detecção automática de sentimentos em comentários nas mídias sociais [dissertação]*. [S.l.]: Belo Horizonte: Fundação Mineira de Educação e Cultura, 2017.

BRUM, H. B.; NUNES, M. d. G. V. Building a sentiment corpus of tweets in brazilian portuguese. *arXiv preprint arXiv:1712.08917*, 2017.

COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE transactions on information theory*, IEEE, v. 13, n. 1, p. 21–27, 1967.

DOMINGOS, P.; HULTEN, G. Mining high-speed data streams. In: ACM. *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2000. p. 71–80.

DOSCIATTI, M. M. Um método para a identificação de emoções básicas em textos em português do brasil usando máquinas de vetores de suporte em solução multiclasse. 2015.

DOSCIATTI, M. M.; FERREIRA, L. P. C.; PARAISO, E. C. Anotando um corpus de notícias para a análise de sentimentos: um relato de experiência (annotating a corpus of news for sentiment analysis: An experience report). In: *Proceedings of the 10th Brazilian Symposium in Information and Human Language Technology*. [S.l.: s.n.], 2015. p. 121–130.

EKMAN, P.; FRIESEN, W. V. Facial action coding system. *Cognition & emotion*, Taylor & Francis, 1978.

EUGENIO, B. D.; GLASS, M. The kappa statistic: A second look. *Computational linguistics*, MIT Press, v. 30, n. 1, p. 95–101, 2004.

FEHR, B.; RUSSELL, J. A. Concept of emotion viewed from a prototype perspective. *Journal of experimental psychology: General*, American Psychological Association, v. 113, n. 3, p. 464, 1984.

FOUAD, M. M.; GHARIB, T. F.; MASHAT, A. S. Efficient twitter sentiment analysis system with feature selection and classifier ensemble. In: HASSANIEN, A. E. et al. (Ed.). *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018)*. Cham: Springer International Publishing, 2018. p. 516–527.

FREITAS, C. et al. Sparkling vampire... lol! annotating opinions in a book review corpus. *New Language Technologies and Linguistic Research: A Two-Way Road. Cambridge Scholars Publishing*, p. 128–146, 2014.

GAZZANIGA, M.; HEATHERTON, T. Ciência psicológica: Mente, cérebro e comportamento (mav veronese, trans.). *Porto Alegre: Artmed.(Trabalho original publicado em 2005)*, 2007.

GROSSMAN, D. A.; FRIEDER, O. *Information retrieval: Algorithms and heuristics*. [S.l.]: Springer Science & Business Media, 2012.

HAN, J.; PEI, J.; KAMBER, M. *Data mining: concepts and techniques*. [S.l.]: Elsevier, 2011.

INDURKHYA, N.; DAMERAU, F. J. *Handbook of natural language processing*. [S.l.]: CRC Press, 2010.

IZARD, C. E. The face of emotion. Appleton-Century-Crofts, 1971.

JACK, R. E.; GARROD, O. G.; SCHYNS, P. G. Dynamic facial expressions of emotion transmit an evolving hierarchy of signals over time. *Current biology*, Elsevier, v. 24, n. 2, p. 187–192, 2014.

JANSEN, B. J. et al. Micro-blogging as online word of mouth branding. In: ACM. *CHI'09 extended abstracts on human factors in computing systems*. [S.l.], 2009. p. 3859–3864.

JIANG, S. et al. An improved k-nearest-neighbor algorithm for text categorization. *Expert Systems with Applications*, Elsevier, v. 39, n. 1, p. 1503–1509, 2012.

JURAFSKY, D.; MARTIN, J. H. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. [S.l.]: Prentice Hall, Pearson Education International, 2009. 1–1024 p.

KOURTELLIS, N. et al. Vht: Vertical hoeffding tree. In: IEEE. *Big Data (Big Data), 2016 IEEE International Conference on*. [S.l.], 2016. p. 915–922.

LANG, P. J. The emotion probe: studies of motivation and attention. *American psychologist*, US: American Psychological Association, v. 50, n. 5, p. 372, 1995.

LI, F. A Pattern Query Strategy Based on Semi-supervised Machine Learning in Distributed WSNs. *Journal of Information and Computational Science*, v. 11, n. 18, p. 6447–6459, 2014.

LIU, B. Sentiment analysis and subjectivity. *Handbook of natural language processing*, v. 2, p. 627–666, 2010.

LIU, B. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, Morgan & Claypool Publishers, v. 5, n. 1, p. 1–167, 2012.

LORENA, A. C. *Investigação de estratégias para a geração de máquinas de vetores de suporte multiclasses*. Tese (Doutorado) — Universidade de São Paulo, 2006.

MARTINS, T. M. R. *Tendências no Twitter*. Tese (Doutorado), 2014.

MCDOUGALL, W. *An introduction to social psychology*. [S.l.]: Psychology Press, 2015.

MEDHAT, W.; HASSAN, A.; KORASHY, H. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, Elsevier, v. 5, n. 4, p. 1093–1113, 2014.

MEULEMAN, B.; SCHERER, K. R. Nonlinear appraisal modeling: An application of machine learning to the study of emotion production. *IEEE Transactions on Affective Computing*, IEEE, v. 4, n. 4, p. 398–411, 2013.

MIKOLOV, T. et al. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

MITCHELL, T. M. *Machine learning*. McGraw-Hill, 1997. (McGraw Hill series in computer science). ISBN 978-0-07-042807-2. Disponível em: <http://www.worldcat.org-/oclc/61321007>.

MORAES, R.; VALIATI, J. F.; NETO, W. P. G. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, Elsevier, v. 40, n. 2, p. 621–633, 2013.

MORALES, G. D. F.; BIFET, A. Samoa: scalable advanced massive online analysis. *Journal of Machine Learning Research*, v. 16, n. 1, p. 149–153, 2015.

MOWRER, O. *Learning theory and behavior. Hoboken, NJ, US.* [S.l.]: John Wiley & Sons Inc. http://dx. doi. org/10.1037/10802-000, 1960.

MUNEZERO, M. D. et al. Are they different? affect, feeling, emotion, sentiment, and opinion detection in text. *IEEE transactions on affective computing*, IEEE, v. 5, n. 2, p. 101–111, 2014.

MURDOPO, A. Distributed decision tree learning for mining big data streams. *Master of Science Thesis, European Master in Distributed Computing*, 2013.

NASCIMENTO, P. et al. Análise de sentimento de tweets com foco em notícias. In: *Brazilian Workshop on Social Network Analysis and Mining*. [S.l.: s.n.], 2012.

ORTONY, A.; NORMAN, D. A.; REVELLE, W. Affect and proto-affect in effective functioning. *Who needs emotions*, p. 173–202, 2005.

PAK, A.; PAROUBEK, P. Twitter as a corpus for sentiment analysis and opinion mining. In: *LREc*. [S.l.: s.n.], 2010. v. 10, n. 2010, p. 1320–1326.

PANG, B.; LEE, L. et al. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, Now Publishers, Inc., v. 2, n. 1–2, p. 1–135, 2008.

PICARD, R. W. et al. Affective computing. Perceptual Computing Section, Media Laboratory, Massachusetts Institute of Technology, 1995.

PORIA, S. et al. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, Elsevier, v. 37, p. 98–125, 2017.

RAHNAMA, A. A. Real-time sentiment analysis of twitter public stream. 2015.

RAMOS, J. et al. Using tf-idf to determine word relevance in document queries. In: *Proceedings of the first instructional conference on machine learning*. [S.l.: s.n.], 2003. v. 242, p. 133–142.

RAVI, K.; RAVI, V. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, Elsevier, v. 89, p. 14–46, 2015.

REIS, J. C. et al. Uma abordagem multilıngue para análise de sentimentos. In: *IV Brazilian Workshop on Social Network Analysis and Mining (BraSNAM 2015)*. [S.l.: s.n.], 2015.

SERRANO-GUERRERO, J. et al. Sentiment analysis: A review and comparative analysis of web services. *Information Sciences*, Elsevier, v. 311, p. 18–38, 2015.

SILVA, I. R. da et al. Classifying emotions in twitter messages using a deep neural network. In: SPRINGER. *International Symposium on Distributed Computing and Artificial Intelligence*. [S.l.], 2018. p. 283–290.

SILVA, I. S. et al. Análise adaptativa de fluxo de sentimento baseada em janela deslizante ativa. In: *SBBD (Short Papers)*. [S.l.: s.n.], 2011. p. 49–56.

SILVA, I. S. et al. Effective sentiment stream analysis with self-augmenting training and demand-driven projection. In: ACM. *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. [S.l.], 2011. p. 475–484.

SPECIA, L.; NUNES, M. d. G. V. *Desambiguação lexical automática de sentido: um panorama*. [S.l.]: ICMC-USP, 2004.

STOJANOVSKI, D. et al. Deep neural network architecture for sentiment analysis and emotion identification of twitter messages. *Multimedia Tools and Applications*, Springer, p. 1–30, 2018.

SU, J.; ZHANG, H. A fast decision tree learning algorithm. In: *AAAI*. [S.l.: s.n.], 2006. v. 6, p. 500–505.

THELWALL, M.; BUCKLEY, K.; PALTOGLOU, G. Sentiment in twitter events. *Journal of the American Society for Information Science and Technology*, Wiley Online Library, v. 62, n. 2, p. 406–418, 2011.

THET, T. T.; NA, J.-C.; KHOO, C. S. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of information science*, Sage Publications Sage UK: London, England, v. 36, n. 6, p. 823–848, 2010.

TSYTSARAU, M.; PALPANAS, T. Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery*, Kluwer Academic Publishers, v. 24, n. 3, p. 478–514, 2012.

VAPNIK, V. N.; CHERVONENKIS, A. Y. On the uniform convergence of relative frequencies of events to their probabilities. In: *Theory of Probability and its Applications*. [S.l.: s.n.], 1971. p. 283–305.

WATSON, J. B. Behaviorism, rev. WW Norton & Co, 1930.

WITTEN, I. H. et al. *Data Mining: Practical machine learning tools and techniques*. [S.l.]: Morgan Kaufmann, 2016.

YE, J. et al. Stochastic gradient boosted distributed decision trees. In: ACM. *Proceedings of the 18th ACM conference on Information and knowledge management*. [S.l.], 2009. p. 2061–2064.

Apêndices

A

Classificação em Flúxo Contínuo de Dados

Nesta parte do trabalho, está descrita o estudo realizado no início desta pesquisa.

Os dados gerados pela plataforma *Twitter* tem potencial infinito, tendo em vista que dependendo do domínio de estudo, o fluxo de *tweets* gerados, possuem um fluxo de produção muito grande. Para isso seria necessário o uso de técnicas de Flúxo Contínuo de Dados - FCD.

Alguns trabalhos relatam estudos de FCD no *Twitter*.

Em (JANSEN et al., 2009) uma análise de comentários sobre algumas marcas com o objetivo de verificar se micro-blogs podem ser considerados mecanismos para propaganda online, sendo verificado que estes micro-blogs podem ser utilizados tanto para divulgar como para obter informações dos clientes. No trabalho de (BIFET; FRANK, 2010) apresentam uma discussão sobre os desafios da mineração em fluxo de dados do Twitter. Em (SILVA et al., 2011b) e (SILVA et al., 2011a) apresenta um classificador baseado em regras de associação com uma solução de esquecimento baseada em Janela de treino Deslizante Ativa. A abordagem proposta foi avaliada experimentalmente a partir da simulação da classificação em tempo real de mensagens enviadas através do *Twitter*.

Nesse contexto, estudos preliminares pretendiam tratar questões de abordadas nesta dissertação abordando questões de FCD, no entanto os dados obtidos para análise, não atendiam as necessidades mínimas para construção de um cenário de FCD, principalmente no quisito quantidade de dados para simular flúxo adequado para os algoritmos escolhidos para tratar questões de FCD em dados textuais. Esse foi um dos motivos que fizeram com que esses estudos não entrassem no escopo principal destes trabalho e ficaram como trabalhos futuros. Um outro motivo, foi que os resultados obtidos com o cenário construídos para FCD, ps resultados não foram promissores, sendo inviável até uma possível comparação com os mostrados na pesquisa.

Os algoritmos selecionados para trabalhar, são baseados em árvores de decisão. A seguir serão descritos tais algoritmos.

A.1 Árvores de Decisão

O aprendizado em árvores de decisão é um dos métodos mais utilizados e práticos para inferência indutiva. É um método para aproximar funções de valores discretos, que são robustas em dados ruidosos e capazes de aprender expressões disjuntivas (MITCHELL, 1997).

A indução da árvore de decisão é a aprendizagem das árvores de decisão a partir das tuplas de treinamento rotuladas. É uma estrutura de árvore em que cada nó interno (nó não-folha) denota um teste em um atributo, cada ramo representa uma saída do teste e cada nó folha (ou nó terminal) possui um rótulo de classe. O nó mais alto de uma árvore é o nó raiz (HAN; PEI; KAMBER, 2011).

De acordo com (SU; ZHANG, 2006) devido ao vasto espaço de busca, o aprendizado de árvores de decisão geralmente é um processo ganancioso, de cima para baixo e recursivo, começando com todos os dados de treinamento e uma árvore vazia. O atributo que melhor representar a partição de dados de treinamento é escolhido como atributo de divisão para a raiz, e os dados de treinamento são então particionados em subconjuntos disjuntos que satisfazem os valores do atributo de divisão. Para cada subconjunto, o algoritmo prossegue repetidamente até que todas as instâncias de um subconjunto pertençam a mesma classe.

Existem medidas de seleção de atributos, que é a heurística utilizada para selecionar o critério de divisão que melhor particiona os dados de treinamento, tais como Ganho de Informação (*Information Gain*), Razão de Ganho (*Gain Ratio*) e Indice Gini (*Gini Index*).

A medida Ganho de Informação é usada pelo clássico algoritmo de árvore de decisão ID3 (*Iterative Dichotomiser*), proposto em 1986, por J. Ross Quilan. Essa medida é baseada na teoria da informação, trabalho de Claude Shannon. Tal abordagem minimiza o número esperado de testes necessários para classificar uma determinada tupla e garante que uma árvore simples, não necessariamente a mais simples, seja encontrada.

O ganho de informação é um método baseado na entropia, que é uma medida que define a pureza de um conjunto de instâncias. Dessa forma, o ganho de informação para um atributo A de um conjunto de dados S evidencia a medida na diminuição da entropia esperada quando é utilizado o atributo A para realizar a partição do conjunto de dados. A entropia é dada por:

$$\text{Entropia(A)} = \sum_{i=1}^{c} -p_1 \log_2 p_i \tag{A.1}$$

Onde p_i é a porção de S pertencente a classe i e c é o conjunto de classes do conjunto

E o ganho de informação será dado por:

$$\text{Ganho(S,A)} = \text{Entropia(S)} - \sum_{v \epsilon A} \frac{|S_v|}{|S|} \text{ Entropia}(S_v) \tag{A.2}$$

Onde S_v é o subconjunto de S formado por exemplos em que o atributo A tem valor v.

Tendo por base as Equações 4.2 e 4.3 é possível realizar o cálculo do ganho de informação, e assim, ter um conjunto de dados bem melhor particionado, dando uma maior precisão na representação final da indução da árvore.

A.2 Árvore de Hoeffding

No trabalho de (DOMINGOS; HULTEN, 2000) foi proposto o conceito de Árvore de Hoeffding (AH), que é um algoritmo de árvore de decisão em fluxo de dados, de distribuição estática e potencialmente infinito. Tendo como princípio básico a AH, os autores propuseram o algoritmo *Very Fast Decison Tree (VFDT)* que em sua estrutura usa o conceito de AH.

Há dois pontos importantes na AH que devem ser ressaltados: o primeiro esta no fato de que cada exemplo que chega é processado uma única vez, e segundo é o Limiar de Hoeffding (LH), que é usado para decidir quantos exemplos serão necessários para que ocorra a divisão de uma folha para criação de um nó na árvore. O LH estabelece que: dada uma variável r de valor real e aleatório, de domínio R, seja n o número de observações independentes dessa variável, de média r. Dessa forma, o LH afirma com probabilidade 1 - σ que a verdadeira média de r é r- ϵ, sendo ϵ definido por:

$$\epsilon = \sqrt{\frac{R^2 \ln (1/\sigma)}{2n}} \tag{A.3}$$

Com base na AH (DOMINGOS; HULTEN, 2000) propuseram o VFDT, um algoritmo incremental, com tempo de processamento constante dos exemplos, que por meio de uma heurística G, Ganho de Informação ou Índice Gini, compara os dois melhores atributos para definição de instalação do teste para a criação de um novo nó na árvore. Os dados são processados um a um de acordo com sua chegada pelo fluxo. A cada novo exemplo, a medida heurística é calculada e os valores dos dois melhores atributos são comparados. Se a diferença entre eles não satisfizer o teste estatístico (nesse caso o LH) o VFDT continua analisando mais exemplos até que haja evidências estatísticas suficientes para que um dos atributos seja escolhido.

A principal característica do VFDT é o uso do LH que é usado para decidir quantos exemplos devem ser analisados antes da instalação de um teste de divisão em um nó. Aos

modelos que usam o LH atribui-se o nome de Árvores de Hoeffding. A seguir será mostrada a modelagem que usa o LH.

Considere G, uma heurística usada para seleção do melhor teste para um nó e que X_a e X_b são os dois atributos de melhores valores para G. Seja $\Delta\ G_r$ = G(X_a) - G(X_b) a diferença dos valores das heurísticas dos atributos, uma vez que o algoritmo usa todo o conjunto de dados, e $\Delta\ \bar{G}_r = \bar{G}(X_a)$ - $\bar{G}(X_b)$ seja a diferença observada quando n exemplos chegam a um nó.

Por meio do LH, sabe-se que com probabilidade 1 - σ que $\Delta\ G_r > \Delta$G – ϵ. Sendo $\Delta\ \bar{G} > \epsilon$, logo é possível afirmar que ΔGr > 0, então G(X_a) > G(X_b), com probabilidade , e, por1 - σtanto, que X_a é o melhor atributo para o nó interno de uma folha. O valor de δ é determinado pelo usuário.

Em outras palavras, para afirmar que o atributo X_a é a melhor escolha que o X_b é necessário a verificação se ΔG > ϵ. Dai deve-se processar exemplos até que ocorra ΔG > ϵ, para o valor escolhido de δ e o número corrente de exemplos em uma folha.

Esse processo garante que a árvore induzida é assintoticamente semelhante a uma onde o teste de divisão seja feito com a inspeção de todo o conjunto de treinamento.

Os elementos a seguir destacam a principal contribuição computacional do VFDT quando trabalha-se com fluxo de dados, onde os mesmos chegam de maneira constante e são potencialmente infinitos:

- Quando dois atributos possuem valores muito próximos, o algoritmo pode ter problemas para eleger o melhor entre eles, e é custoso ficar processando os dados, pois são necessários muitos atributos para essa decisão, para isso um parâmetro de desempate τ é usado, onde ΔG < ϵ < τ, sendo τ um valor especificado pelo usuário;

- O cálculo da heurística G, que contribui para a redução do tempo de processamento de um exemplo. Pois é definido um número mínimo de exemplos e_{min} que uma folha deve ter acumulado antes de recalcular G, pois é inviável o cálculo de G para cada novo exemplo;

- O uso de contadores permite o armazena apenas das heurísticas obtidas, sendo feito o descarte dos dados já processados, não havendo necessidade de armazená-los na memória RAM, uma vez que nem seria algo viável, pois os dados possuem tamanho potencialmente infinito. Quando a memória disponível é atingida, o algoritmo desativa as folhas menos significantes para que seja feita a liberação de espaço;

- O algoritmo pode receber uma árvore já inicializada como partida para a construção da árvore.

A.3 Árvore de Decisão Paralela

Uma das principais desvantagens das árvores de decisão ao trabalhar com um grande volume de dados é que elas precisam processar uma grande quantidade de atributos, o que causa uma sobrecarga computacional. Árvores de decisão paralelas ou distribuídas, realizam a classificação antecipada, ou distribuída para amenizar tais desvantagens (RAHNAMA, 2015).

Na tentativa de minimizar as desvantagens das tradicionais árvores de decisão, quando o domínio é de grande quantidade de dados, algumas propostas surgem(BEN-HAIM; TOM-TOV, 2010) que propõem um algoritmo para construção de árvores de decisão paralela. O uso do paralelismo horizontal na distribuição dos atributos e a construção da árvore é feita a partir dos histogramas dos dados que são mantidos nos processos de trabalho. No entanto, por usar divisão horizontal, há uma sobrecarga de memória do modelo replicado, o que representa um problema para a escalabilidade.

O trabalho de (YE et al., 2009), mostra como distribuir e paralelizar por meio do algoritmo *Gradient Boosted Decision Tree* - (GBDT). A implementação inicial é baseada no MapReduce, com divisão horizontal dos dados. Porém há uma sobrecarga no HDFS - (*Hadoop Dristributed File System*), sistema de distribuição dos dados, responsável pela comunicação ao dividir nós deixando evidente que o MapReduce não é adequado para o algoritmo. Além da questão do HDSF, o tipo de particionamento usado também representa um problema, então o algoritmo é implementado usando particionamento vertical e usando MPI (*Message Passing Interface*), que é um padrão de comunicação de dados em computação paralela.

Em seu trabalho (MURDOPO, 2013) (Arinto Murdopo, 2013) destaca dois tipos de paralelismo, vertical e horizontal. E diz que árvores de decisão paralelas são bem eficientes em cenários de mineração de texto, principalmente quando é utilizado paralelismo vertical, pois é bastante apropriado para tratar domínios com um número elevado de atributos.

A.3.1 Tipos de Paralelismo

No AM, os dados normalmente são representados na forma de uma matriz de dimensão *mxn*, em que m é o número de instâncias de dados e n o número de atributos do conjunto de dados conforme mostra a Figura A.1. No entanto, é necessário encontrar um vetor x, para melhor manipular esses dados.

De acordo com (KOURTELLIS et al., 2016) existem duas maneiras de particionar a matriz de dados para obter paralelismo de dados: por linha ou por coluna. O primeiro é chamado de paralelismo horizontal e o último de paralelismo vertical, conforme mostra a Figura A.2. Com o paralelismo horizontal, as instâncias de dados são independentes umas das outras e podem ser processadas isoladamente enquanto consideram todos os

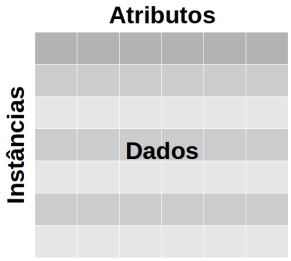


Figura A.1 – Matriz de dados mxn.

atributos. Já no paralelismo vertical, os atributos são considerados independentes uns dos outros.

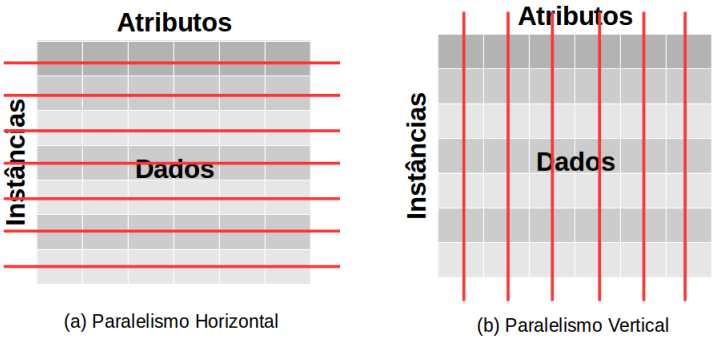


Figura A.2 – Tipos de Paralelismo.

O paralelismo horizontal é indicado em cenários que possuem taxas de chegada bastante altas, onde pode ser adicionado mais poder de processamento para lidar com essas instâncias de chegada. Entretanto, esse tipo de paralelismo necessita de uma quantidade grande de memória, e por gastar maior parte do seu tempo de processamento com o cálculo do ganho de informação, não é indicado para situações onde as instâncias possuem grande quantidade de atributos.

O paralelismo vertical assim como o horizontal, também gasta maior parte do seu processamento no cálculo do ganho de informação, mas pela natureza do seu particionamento, é indicado para casos em que há um elevado número de atributos. Esse cenário é bem propício para mineração de texto, pois os algoritmos de mineração de texto normalmente trabalham com entradas 10000 a 50000 em seus dicionários. As entradas de texto no algoritmo são transformadas em instâncias, e cada palavra do texto corresponde a um atributo.

A.3.2 Vertical Hoeffding Tree

No trabalho de (MORALES; BIFET, 2015) os autores definem o *Vertical Hoeffding Tree* - VHT como uma extensão distribuída do VFDT de Domingos e Hulten. O VHT usa paralelismo vertical para dividir a carga de trabalho em várias máquinas. O

paralelismo vertical alavanca o paralelismo entre atributos no mesmo exemplo, e não em diferentes exemplos no fluxo. Na prática, cada exemplo de treinamento é roteado pelo modelo da árvore para uma folha. Lá, o exemplo é dividido em seus atributos constituintes, e cada atributo é enviado para uma instância de processamento diferente, que mantém as estatísticas suficientes.

O VHT tem como vantagem o uso reduzido de memória, uma vez que os contadores de atributos não são replicados nas diversas máquinas. O algoritmo faz o processamento dos atributos de maneira paralela para que seja determinado o melhor atributo local, para dividir e combinar os resultados da computação paralela e poder decidir o melhor atributo global para divisão e a árvore poder crescer.

O algoritmo VHT pode ser representado conforme mostra o diagrama da Figura A.3.

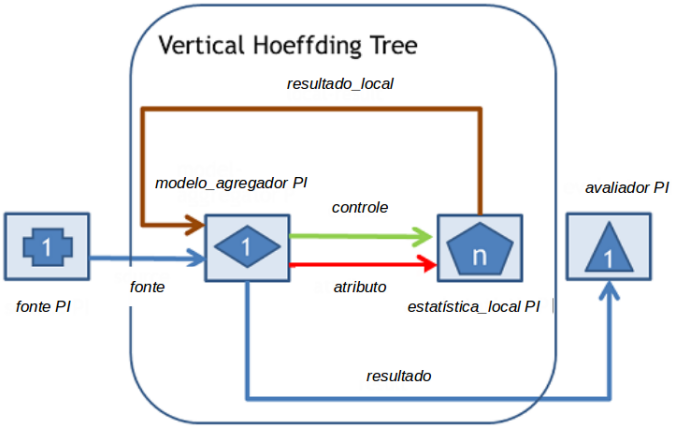


Figura A.3 – Diagrama de processo do VHT, (adaptado de (LI, 2014)).

Cada um dos quadrados do diagrama da Figura A.3 equivale a um Item de Processamento (PI), que representa uma unidade de elemento computacional, que pode ser um nó, um segmento de processo, que executa uma parte do algoritmo. Podemos observar também os seguintes componentes do VHT, *fonte* PI, *modelo_agregador* PI, *estatística_local* PI, *avaliador* PI, *resultado_local*. O número presente nos quadrados indica o nível de paralelismo em cada um deles. O nível de paralelismo do algoritmo é definido pelo número de *estatísticas_local* PI.

O algoritmo está dividido em dois componentes principais: *modelo_agregador* PI e *estatística_local* PI.

O *modelo_agregador* PI é o componente que mantém o modelo atualizado da árvore produzida até o momento. A *fonte* PI envia as instâncias de chegada para o *modelo_agregador* PI para realizar a classificação e enviá-las para a folha correta. As instâncias de chegada que não estiverem rotuladas, o modelo prediz um rótulo e as envia para avaliação. Caso estejam com rótulo, elas são usadas como dados de treinamento.

A *estatística_local* PI é uma estrutura de tabela distribuída, que acumula as estatísticas suficientes para um atributo de classe n_{ijk} enviados pelo *modelo_agregador* PI. A indexação da tabela é feita por ID_folha e ID_coluna e cada uma das células representa o conjunto de contadores, um para cada par de valor de atributo e classe

O *modelo_agregador* PI se comunica com a *estatística_local* PI enviando instâncias de divisão por um fluxo de atributos e, por meio do fluxo de controle, solicita a computação das estatísticas locais dos atributos enviados. Já os resultados da computação das *estatísticas_locais* PI são enviados por meio do fluxo de *resultados_locais*.

O processo descrito acima caracteriza a fase de aprendizagem do VHT, que é semelhante a do VFDT. O que difere é o envio de atributos para as *estatísticas_locais* PI para serem processadas e a existência de uma lista de folhas de divisão, que armazena os atributos candidatos para divisão.

Além da fase de aprendizagem, o VHT também tem a fase de crescimento, onde cada uma das *estatísticas_locais* PI realiza o cálculo do Ganho de Informação e seleciona os dois melhores atributos locais e os envia para o *modelo_agregador* PI.

A partir de então, tendo posse dos valores dos dois melhores atributos locais de todas as *estatísticas_locais* PI, o *modelo_agregador* PI decide se divide ou não a folha para criar um novo nó. Essa decisão ocorre por dois motivos:

- Quando as condições $\Delta\ \bar{G} > \epsilon$ e $\epsilon < \tau$, originada do VFDT, forem satisfeitas;
- Ou quando o tempo limite para que haja a escolha dos dois melhores atributos seja atingido. Limite esse definido no *modelo_agregador* PI.

É importante destacar que todos os cálculos realizados nessa fase são os mesmos realizados no VFDT.

O avaliador PI é o responsável pela avaliação da classificação enviada pelo modelo agregador PI.

Essa avaliação ocorre em termos de precisão ou taxa de transferência.

A.4 Experimentos

Foram realizados experimentos com a base citada no Capítulo 3 deste trabalho, usando os mesmos processos de tratamento e vetorizando os dados com uso de TF - IDF e Word2Vec, porém a acurácia obtida para ambos os algoritmos (VFDT e VHT) foram muito baixos, e por isso não são apresentados. Pois no VFDT não foi possível determinar a precisão e a F1 - Score dos dados classificados, dessa forma não seria possível garantir o quão foi boa ou ruim a classificação.

Para o VHT, os dados não permitiram poder testar cenários com diferentes níveis de paralelismo, o que é um diferencia do VHT, sobretudo para uma grande quantidade de dados a serem processados.

No entanto, as pesquisas e estudos realizados com esses algoritmos, representam uma possibilidade de trabalhos futuros, para o aprimoramento dos dados, ou a aplicação de outros dados, com outro domínio, na tarefa de classificação multiclasse de dados textuais para a tarefa de Análise de Sentimentos.