Caio Magno Aguiar de Carvalho

Estudo Comparativo de Análise de Sentimento Aplicado à Notícias Políticas

São Luís - Maranhão - Brasil

20/02/2018

Caio Magno Aguiar de Carvalho

Estudo Comparativo de Análise de Sentimento Aplicado à Notícias Políticas

Dissertação apresentada ao curso de Mestrado em Engenharia Elétrica da Universidade Federal do Maranhão, como requisito parcial para a obtenção do grau de mestre.

Universidade Federal do Maranhão – UFMA

Departamento de Engenharia Elétrica

Programa de Pós-Graduação

Orientador: Allan Kardec Barros

Coorientador: Ewaldo Santana

São Luís - Maranhão - Brasil

20/02/2018

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Aguiar de Carvalho, Caio Magno.
Estudo Comparativo de Técnicas de Análise de Sentimento Aplicado à Notícias Políticas / Caio Magno Aguiar de Carvalho. - 2018.
49 f.

Orientador(a): Allan Kardec Duailibe Barros Filho.
Dissertação (Mestrado) - Programa de Pós-graduação em Engenharia de Eletricidade/ccet, Universidade Federal do Maranhão, São Luís - Maranhão, 2018.

1. Análise de Sentimento. 2. Aprendizagem de Máquina. 3. Processamento de Linguagem Natural. 4. Seleção de atributos. I. Duailibe Barros Filho, Allan Kardec. II. Título.

Caio Magno Aguiar de Carvalho

Estudo Comparativo de Análise de Sentimento Aplicado à Notícias Políticas

Dissertação apresentada ao curso de Mestrado em Engenharia Elétrica da Universidade Federal do Maranhão, como requisito parcial para a obtenção do grau de mestre.

Trabalho aprovado. São Luís - Maranhão - Brasil, 24 de novembro de 2012:

Allan Kardec Barros
Orientador

Ewaldo Santana
Orientador

João Viana Fonseca
Universidade Federal do Maranhão

Paulo Armando Cavalcante Aguilar
Universidade Federal do Ceará

São Luís - Maranhão - Brasil
20/02/2018

Para Aline Késsia
esposa, companheira e conselheira

Agradecimentos

Ao professor Allan Kardec, não somente pela oportunidade de trabalhar no PIB como também pelos valiosos *insights*. Ao professor Ewaldo Santana, que me ensinou (e continua ensinando) a importância da objetividade no meio científico. À Hitoshi Nagano, por toda paciência e disposição em ajudar e ensinar a trilha da Aprendizagem de Máquina e do Processamento de Linguagem Natural.

À Gean Carlos e Jonatham Queiroz pelas lições e conselhos dados na área de estatística e processamento de sinais. À Daniel Luna, como companheiro no aprendizado nessa fase e colaborador nos estudos de aprendizagem de máquina. A todos os outros amigos do laboratório de Processamento da Informação Biológica que tornaram esse caminho mais aprazível através dos ótimos momentos compartilhados.

"Quem pensa conhecer alguma coisa, ainda não conhece como deveria."
1 Coríntios 8:2

Resumo

No período eleitoral, grande parte da opinião pública sobre partidos e candidatos é formada a partir de notícias veiculadas através dos meios de comunicação de massa: TV, radio, jornal e principalmente internet, através de portais de notícias online. Entretanto, existe um debate sobre a verdadeira imparcialidade desses meios ao transmitir a informação aos telespectadores. Alguns acusam a mídia de favorecer algumas figuras políticas e suas agendas, enquanto outros reafirmam a imparcialidade deste meio de comunicação. Entretanto, julgar a parcialidade de notícias políticas é uma tarefa que está sujeita a subjetividade do avaliador, que nem sempre reflete a realidade. Neste contexto, os métodos providos pelo Processamento de Linguagem Natural, através do campo de estudo da Análise de Sentimento, podem trazer uma visão menos enviesada nessa discussão. Análise de Sentimento é campo que alia as ferramentas de mineração de texto com ferramentas aprendizagem de máquina afim de classificar textos de acordo com sentimento expresso: positivo, negativo ou neutro. Neste trabalho é proposto um estudo comparativo entre as técnicas de representação de texto, seleção de atributos e ferramentas de aprendizagem de máquina para se classificar notícias políticas coletadas em portais online sobre as eleições brasileiras de 2014 quanto a sua opinião/sentimento (positivo, negativo ou neutro). Neste estudo os classificadores Naïve Bayes, *Support Vector Machines* e Regressão Logística (ou MaxEnt) são avaliados juntamente com as técnicas de seleção de atributos Qui Quadrado, *Categorical Proportional Difference* e *Categorical Probability Proportional Difference*. Os experimentos conduzidos visam escolher a melhor representação vetorial do texto, o melhor método de seleção de atributos e o melhor classificador para a base de dados proposta. A avaliação é realizada através de validação cruzada medindo-se a acurácia média e seu desvio-padrão para cada experimento. Os resultados experimentais apontam para representação *bag-of-words* utilizando vocabulário de *unibigrams* selecionados pela técnica *Categorical Probability Proportional Difference* juntamente com o classificador MaxEnt, atigindo uma acurácia média de 84,45% com um desvio-padrão de 0.029.

Palavras-chaves: Processamento de Linguagem Natural. Análise de Sentimento. Aprendizagem de Máquina. Seleção de atributos.

Abstract

In the elections period, the public opinion about parties and candidates is partially influenced by mainstream media as TV, radio, newspappers and mainly internet through newswire media. However, there is a debate about the impartiality in these media when transmitting news. Sometimes it is acused to favour some political entities and its agendas, while others affirm its neutrality. Assess news article in this context is not a simple task, because the evaluation could be influenced by some biases of who assesses that article. The methods provided by Natural Language Processing, through the field of Sentiment Analysis, could bring a less biased viewpoint of that question. Sentiment Analysis joins text mining techiniques and machine learning tools to classify texts according its sentiment polarity (positive, negative or neutral).In this work we propose a comparative study between sentiment analysis text representation models, feature selection techiniques and machine learning classifiers in order to classify the polarity of political online news about 2014 brazilian elections. In this study the classifiers Naïve Bayes, Support Vector Machine and Logistic Regression (MaxEnt) are evaluated with feature selection techinques as Chi Square, Categorical Proportional Difference, Categorical Probability Propotional Difference. The experiments sought to choose the best text representation, feature selection techinique and machine learning classifier. The evaluation is made by cross validation measuring accuracy mean and its standard deviation. The experimental results pointed to the bag-of-words representation with unibigram selected by Categorical Probability Proportional Difference with MaxEnt classifier achieving 84.45% with standard deviation of 0.029.

Key-words: latex. abntex. text editoration.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

UCG	*User Generated Content*
NLP	*Natural Language Processing*
SVM	*Support Vector Machine*
MaxEnt	Regressão Logística por modelagem de máxima entropia
CPD	Categorical Proportional Difference
CPPD	Categorical Probability Proportional Difference

Sumário

1 Introdução

A opinião de terceiros geralmente é um fator importante na tomada de decisões como o que comer, qual roupa usar, qual é a melhor oficina do bairro ou qual é o político mais confiável para se votar. Esse processo de troca de informação foi potencializado pelo surgimento da internet ao aumentar o alcance da comunicação e pela enorme quantidade de informação disponível e facilmente acessível. Dessa forma, opiniões e comentários provindos de populações mais diversificadas, acerca do consumo de bens e serviços oferecidos ao público podem ser consultados na rede por qualquer pessoa conectada.

Estas avaliações encontrados na internet têm se mostrado valiosas tanto para o consumidor como para quem oferece produtos e serviços utilizando a internet como veículo. Os empresários tem visto aí uma oportunidade de aperfeiçoar e estabelecer suas marcas através do contato direto com a opinião pública. Os usuários avaliam diretamente os produtos nos respectivos sites de compras ou através de postagens em redes sociais. A empresa de bebidas Starbucks é um exemplo: a empresa utilizou os dados acerca da preferência dos pedidos de seus clientes para lançar bebidas enlatadas e engarrafadas de modo que pudessem abranger a preferência da maior parte de seus consumidores [1].

Percebendo o potencial desse mecanismo, diversas empresas procuram investir em tecnologias de mineração de dados para monitorar atividade do consumidor em relação a sua marca ou produto. Técnicas de Processamento de Linguagem Natural (Natural Language Processing – NLP) e de Aprendizagem de Máquina geralmente têm sido utilizadas como ferramentas para auxiliar a análise dos dados textuais.

Como exemplo de aplicação de algorimos de NLP aplicados a realidade brasileira, a assessoria de comunicação da prefeitura de São Paulo atribuiu o mérito do aumento de sua popularidade em 2017 às ferramentas de análise de dados (especialmente às baseadas em Análise de Sentimento) dentro das redes sociais [2]. Essa técnica produz *insigths* valiosos ao detectar o nível de aceitação do público em relação a alguma declaração feita. Dessa forma é possível ajustar o discurso de forma a torná-lo mais compreensível, reduzir os impactos negativos e potencializar os positivos. O sucesso dessa estratégia tem chamado a atenção de candidatos em potencial para as eleições de 2018, que começam a considerar a Ciência de Dados uma poderosa aliada de campanha. [3].

Embora as mídias sociais tenham sido o grande alvo dos algoritmos de analise de dados, mídias como os noticiários têm sido deixadas de lado nessa investigação. Apesar

[1] http://www.cnbc.com/2016/04/06/big-data-starbucks-knows-how-you-like-your-coffee.html, acessado em 11/01/2018

[2] http://www.bbc.com/portuguese/brasil-41406420

[3] http://www.bbc.com/portuguese/brasil-41328015

de redes como *Facebook*, *Twitter* e outras serem fontes valiosas para mineração de opinião pública, as notícias também podem ser indicadores interessantes de popularidade, pois possuem um parcela de influência na formação da opinião dentro desses ambientes (RITT; WERLE, 2013).

No Brasil a situação torna-se mais delicada pelo fato dos veículos de mídia serem constantemente acusados de parcialidade em relação a determinadas agendas políticas. Essa discussão torna-se ainda mais acentuada durante o período das eleições. A principal acusação é de que os noticiários tem manipulado a opinião publica a favor ou contra determinados partidos ou candidatos em seus períodos de campanha. Entretanto para que essa afirmação seja isenta de qualquer viés, é necessário que seja aplicado um critério uniforme de avaliação por alguém que não expresse nenhuma preferência política.

Dentro desse contexto, os métodos computacionais providos pelo NLP através das técnicas de Análise de Sentimento, podem contribuir nessa discussão trazendo uma perspectiva menos parcial na avaliação das notícias, classificando-as de acordo com o respectivo sentimento/opinião expresso. Dessa forma podemos verificar como a opinião sobre determinada entidade política se distribui entre os veículos de informação, concluindo se estão favorecendo ou desfavorecendo uma em detrimento de outra.

1.1 Trabalhos Relacionados

O Processamento de Linguagem Natural, é a área da computação que lida com o desenvolvimento de sistemas que compreendem a linguagem natural humana. Em Bird, Klein e Loper (2009) os autores definem linguagem natural como a linguagem usada no dia a dia ao nos comunicarmos com outras pessoas. Várias aplicações desse campo de estudo têm sido desenvolvidas como ferramentas para auxiliar e automatizar determinadas tarefas que exigem análise de texto. Os trabalhos desenvolvidos em Yang et al. (2012) e Iqbal, Fung e Debbabi (2012) apresentam ferramentas de extração de informação para auxiliar investigações policiais, enquanto em Calambás et al. (2015) é proposto um sistema de indexação de documentos para consulta de precedentes criminais. Os trabalhos Maynard e Funk (2011) e Jose e Chooralil (2015) apresentam aplicações que interagem com redes sociais afim de realizar previsões em período de eleições, buscando reconhecer algum tipo de padrão em postagens de usuários. Esses padrões revelam a opinião individual do sujeito em relação à entidade política em questão, sendo classificada como positiva, negativa ou neutra. O método utilizado nos dois últimos trabalhos citados compõe uma sub-área de estudo do NLP conhecida como Análise de Sentimento.

A Análise de Sentimento, como subárea do NLP, propõe a desenvolver sistemas que são capazes de analisar textos e sumarizar sua opinião/sentimento automaticamente em relação a um determinado tópico (PANG; LEE et al., 2008). Este método é bastante

utilizado para classificar opinião de clientes sobre produtos em sites de compras (conhecidos como *product reviews*) e críticas e comentários sobre filmes (conhecidos como *movie reviews*) afim de conhecer a opinião dos usuários, como mostrado em Pang, Lee e Vaithyanathan (2002). Em Pang, Lee et al. (2008), sistemas de recomendação e algoritmos de previsão de vendas são apresentados também como aplicações de análise de sentimento.

Schouten e Frasincar (2016) divide a metodologia da Análise de Sentimento em 3 categorias. A primeira se trata dos métodos baseados em léxicos, como LIWC (TAUSCZIK; PENNEBAKER, 2010) e Wordnet (MILLER, 1995). Essa técnica toma as palavras de um texto e compara com um dicionário dedicado composto por no mínimo duas listas de palavras: uma lista de palavras associadas a sentimentos negativos e outra a sentimentos positivos. A polaridade do texto é atribuída de acordo com a lista que possui a maior quantidade de palavras neste. Os métodos baseados em aprendizagem de máquina supervisionado constituem a maioria, sendo esta a metodologia adotada neste trabalho. Classificadores como Naïve Bayes, *Support Vector Machines* e árvores de decisão são bastante populares nesta abordagem (PANG; LEE; VAITHYANATHAN, 2002). Por último, os métodos de aprendizagem de máquina não-supervisionados, como K-médias e *Latent Dirichlet Allocation* que constitui em sumarizar um texto em função dos tópicos abordados nele (BLEI; NG; JORDAN, 2003).

No contexto de aplicações relacionadas a política, os trabalhos Maynard e Funk (2011) e Jose e Chooralil (2015) realizam Análise de Sentimento para prever o resultado das eleições através da popularidade dos candidatos no Twitter. Os *tweets* são classificados individualmente como positivos, negativos ou neutros de forma automática utilizando o léxico Wordnet. Quanto mais *tweets* positivos um candidato tem associado a seu nome, mais popular ele é e mais provável de ser eleito. Ringsquandl e Petkovic (2013) em seu trabalho extrai tópicos relevantes em tweets de figuras políticas através frequência de sintagmas nominais. O autor verificou que a relação entre os tópicos detectados e o candidato melhora o desempenho da classificação do tweet.

Para o português brasileiro, Pasinato, Mello e ao (2015) realiza tarefa semelhante aos trabalhos anteriormente citados, classificando os tweets em relação à um único candidato à prefeitura no Rio de Janeiro em 2012. O autor utiliza os classificadores Naïve Bayes, Árvore de Decisão e Regressão Logística Multinomial (Máxima Entropia) combinando-os com as representações textuais *Term Frequency*, *Term Frequency Inverse Document Frenquency* e Decomposição em Valores Singulares.

Embora haja interesse crescente em NLP, existe uma carência de pesquisas, ferramentas e recursos humanos para quando se trata da língua portuguesa, especialmente o português brasileiro (PARDO et al., 2010). Alguns do trabalhos aplicados ao português brasileiro utilizam o twitter como fonte primária dos dados, afim de extrair a opinião dos usuários acerca de algum assunto Moraes, Manssour e Silveira (2015). No entanto, poucos

trabalhos avaliam textos jornalísticos. Uma explicação plausível para tal pode ser encontrada em Padmaja, Fatima e Bandu (2013), afirmando que este estilo é mais difícil de ser submetido a tarefas de classificação devido a sua pretensa neutralidade e uniformidade, ou seja, alta semelhança sintática entre si.

A Tabela 1 apresenta 3 trabalhos que abordam o problema de classificação de notícias. Embora nenhum deles trate deles trate especificamente de política, a metodologia apresentada serve de base para trabalhos semelhantes.

Tabela 1 – Tabela comparativa de desempenho em trabalhos relacionados ao contexto de classificação de notícias.

Autor (ano)	**Atributos**	Técnica	Num. classes	Acurácia(%)
Morgado (2012)	*unigrams*	*minimum cuts* léxico *Valence Shifters*	3	56.61
Martinazzo, Dosciatti e Paraiso (2011)	*unigram*	LSA	6	70.5
Alvim et al. (2010)	*bigrams*, *POS tags*	SVM	2	84

Na área jornalística existem alguns trabalhos publicados voltados para o português europeu (MORGADO, 2012) e português brasileiro (DOSCIATTI; FERREIRA; PARAISO, 2013; ALVIM et al., 2010).

O trabalho desenvolvido por Morgado (2012) apresenta uma abordagem simples baseada em léxicos e na influência de sentenças próximas. No entanto, a partir dos resultados apresentados é possível concluir que uma metodologia baseada em léxicos não é apresenta resultados comparáveis às baseadas em aprendizagem de máquina.

A aplicação encontrada em Martinazzo, Dosciatti e Paraiso (2011) apresenta uma abordagem baseada em aprendizagem de máquina utilizando LSA para classificar manchetes de notícias dentro de um espectro de 6 sentimentos. A metodologia utiliza a técnica de redução de dimensionalidade chamada Análise Semântica Latente (*Latent Semantic Analysis* - abreviada por LSA), que é um caso especial da Análise em Componentes Principais. Esta técnica é utilizada para agrupar os textos baseado no cálculo da similaridade entre os vetores que representam os textos. A LSA, embora sofisticada, não foi capaz de apropriado para um problema de Análise de Sentimentos de multiclasse. A LSA também é uma técnica computacionalmente custosa a depender da quantidade de dados utilizados para seu treinamento.

Em Alvim et al. (2010), utiliza *bigrams* e suas *POS-Tags* (rótulos que classificam as palavras em suas classes morfológicas) como atributos de sua representação textual. O classificador SVM é utilizado para classificar os textos em duas classes, positivo e

negativo. Embora se trate de um problema com apenas duas classes, é razoável concluir que as classes morfológicas contribuem positivamente para a classificação, porém o uso indiscriminado destas pode levar a explosão da quantidade de atributos.

Alguns trabalhos concentram-se na construção de bancos de textos anotados de notícias para análise de sentimento. Alguns utilizam os rótulos convencionais, como em Arruda, Roman e Monteiro (2015) (positivo, neutro e negativo), outros, como em Dosciatti, Ferreira e Paraiso (2013), utilizam rótulos personalizados para denotar sentimentos como tristeza, revolta e alegria.

A partir do levantamento bibliográfico realizado percebe-se que poucos trabalhos de Análise de Sentimento no Brasil abordam o domínio das notícias, especialmente sobre política. As aplicações relacionadas a política ficam limitadas a previsões eleitorais realizadas através de redes sociais. Aplicações de classificação de notícias políticas podem ser relevantes no estudo de viés presente em agências de notícias, rastreamento de opinião automático e até sendo possivel avaliar a correlação existente entre as notícias e a opinião pública.

1.2 Objetivos

O trabalho proposto neste documento visa colaborar com as pesquisas de classificação de notícias através da aplicação do método de análise de sentimento utilizando o corpus disponibilizado em (ARRUDA; ROMAN; MONTEIRO, 2015) como base de dados. Nesta pesquisa pretende-se classificar as notícias no período das eleições de 2014 como positivas, negativas ou neutras de acordo com o tópico abordado no texto usando técnicas de Análise de Sentimento.

1.2.1 Objetivos Específicos

Afim de se alcançar o objetivo geral deste trabalho, buscaremos atingir os seguintes objetivos específicos:

- Comparar o desempenho de classificadores baseados em aprendizagem de máquina para textos jornalísticos;
- Verificar se o uso de algoritmos supervisionados de seleção de atributos provoca aumento significativo na performance dos dos classificadores utilizados.

1.3 Contribuições

O presente trabalho contribui com o cenário atual de aplicações de NLP provendo resultados preliminares através da da comparação das ferramentas utilizadas (classifica-

dores e métodos de seleção de atributos) em vista da baixa visibilidade deste tipo de conteúdo dentro do contexto brasileiro de Processamento de Linguagem Natural. A intenção é que os resultados encontrados nesse trabalho possam servir como um teste de *benchmark* para futuros trabalhos que venham abordar essa mesma temática.

1.4 Organização do Trabalho

O presente trabalho é organizado na seguinte ordem: o capitulo 2 aborda os conceitos-chave para compreensão deste trabalho como Processamento de Linguagem Natural, classificação de texto, classificadores baseados em aprendizagem de máquina e métodos de seleção de atributos.

No capítulo 3 é descrita a metodologia empregada na condução dos experimentos: aquisição e armazenamento dos dados, etapas de pré-processamento, treinamento dos classificadores e avaliação dos mesmos.

O capítulo 4 apresenta os resultados obtidos, discute a comparação entre os métodos utilizados e compara com resultados de trabalhos semelhantes já publicados.

O capítulo 5 finaliza este trabalho com as conclusões obtidas, apresenta algumas considerações finais e perspectivas de trabalhos futuros.

2 Fundamentação Teórica

Neste capítulo serão abordados os tópicos básicos necessários para a compreensão do trabalho. Primeiramente será apresentado a representação *bag-of-words* de textos que é a base para aplicações de classificação baseadas em aprendizagem de máquina. Os classificadores utilizados são apresentados neste capítulo bem como uma breve apresentação de seu funcionamento e treinamento. Por fim, os métodos de seleção de atributos são expostos.

2.1 Representação de textos com *bag-of-words*

Todas as técnicas usadas aqui para extrair atributos são baseadas em *ngrams*. Um *ngram* é uma sequência de n palavras extraídas de um determinado texto. Quando apenas uma palavra é tomada por vez, esta sequência é chamada de *unigram*. Sequências de duas palavras do texto tomadas da mesma forma são chamada de *bigram*.

O conjunto que reúne todos os *ngrams* presentes no corpus é chamado vocabulário (v). Para cada elemento do vocabulário v é atribuído um índice i de acordo com a equação 2.1, onde w_i é o *ngram* que possui o índice i e n é o número de *ngrams* existentes no *corpus*.

$$v = \{w_1, w_2, w_3, ..., w_n\}\, i, n \in \mathbb{N} \tag{2.1}$$

O método *bag-of-words* consiste em representar um determinado documento como um vetor que armazena a frequência da ocorrência de cada *ngram* do vocabulário no texto. O vetor é construído a partir da contagem das ocorrências c_i dos *ngrams* do vocabulário v. Cada elemento c_i representa a contagem das ocorrências de w_i no documento d conforme descrito na equação 2.2.

$$d = [c_1, c_2, c_3, ..., c_n]\, i, n \in \mathbb{N} \tag{2.2}$$

2.1.1 Exemplo de representação *bag-of-words* com unigrams

Nesta subseção, um pequeno exemplo de como representar textos com *bag-of-words* de unigrams será dado utilizando base de dados os 4 textos a seguir na Figura 1.

Todas as palavras do vocabulário obtido são associadas a um índice numérico. A partir daí, cada documento dessa base pode ser representado vetorialmente a partir da

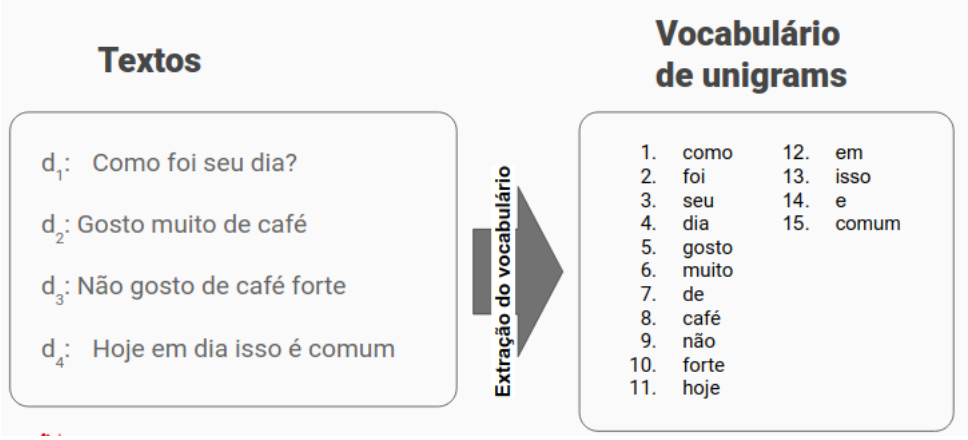

Figura 1 – Ilustração da extração do vocabulário de unigrams a partir de um conjunto de textos

Tabela 2 – Exemplo de crescimento na representação *bag-of-words* de acordo com o tipo de *ngram*

ngram	**Dimensionalidade**
unigram	n
bigram	n^2
trigram	n^3

equação 2.3:

$$D = [C(w_1), C(w_2), ..., C(w_{15})] \tag{2.3}$$

,

onde D é a representação vetorial do texto e $C(w_i)$ é a frequência do unigram cujo indice é i. Dessa forma a frase "Como foi seu dia seria representada pelo seguinte vetor:

$$D = [1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

A dimensionalidade desta representação vetorial depende linearmente do tamanho do vocabulário e cresce exponencialmente ao aumentar-se o tipo de *ngram* (Tabela 2)

Ao aumentar-se essa dimensionalidade através do aumento do *ngram*, os vetores de representação tendem a tornar-se esparsos. Isso se deve ao fato que toda combinação de n palavras será considerada uma dimensão válida nessa representação. Em situações onde a dimensionalidade do vetor é muito alta e esparsa, o modelo aprendido por algoritmos de aprendizagem de máquina pode gerar complexidades desnecessárias, como adaptar-se a uma característica que não é relevante para a distinção dos documentos.

2.2 Classificadores Baseados em Aprendizagem de Máquina

Nesta seção apresentaremos os classificadores Naïve Bayes, Maquina de Vetor de Suporte e Regressão Logística. Estes três classificadores foram escolhidos por conta de sua popularidade em aplicações de Análise de Sentimento (PANG; LEE; VAITHYANATHAN, 2002; SCHOUTEN; FRASINCAR, 2016).

2.2.1 Naïve Bayes

Uma forma de classificar dados textuais é atribuir a um dado documento d a classe c mais provável através do cálculo da probabilidade posterior $c = argmax_c P(c|d)$. O teorema de Bayes da probabilidade condicional permite que esse valor seja calculado a partir de dados previamente rotulados, conforme a Equação 2.5.

$$P(C = c|D = d) = \frac{P(D = d|C = c)P(C = c)}{P(D = d)} \tag{2.4}$$

$$P(C = c|W_1 = w_1..., W_n = w_n) = \frac{P(W_1 = w_1, ..., W_n = w_n|C = c)P(C = c)}{P(W_1 = w_1, ..., W_n = w_n)} \tag{2.5}$$

As probabilidades $P(D = d|C = c)$ podem ser estimadas através através da razão entre o número de documentos d rotulados na classe c. No modelo *bag-of-words* de extração de atributos, o documento d é expresso como um vetor que contém a contagem da ocorrência de cada palavra do vocabulário no documento em questão.

O modelo Naïve Bayes assume que as variáveis aleatórias W_i são independentes entre si, portanto, podemos aplicar a regra da cadeia na Equação 2.5 afim de transformar $P(W_1 = w_1, W_2 = w_2, ..., W_n = w_n)$ no produto $P(W_1 = w_1)P(W_2 = w_2), ..., P(W_n = w_n)$. Dessa forma podemos calcular a probabilidade posterior $P(C = c|D = d)$ através da equação 2.6

$$P(C = c|D = d) =\propto P(C = c)(\Pi_{i=0}^{m} P(W_i = w_i|C = c)) \tag{2.6}$$

.

O denominador $\Pi_{i=0}^{m} P(W_i = wi)$ foi suprimido na Equação 2.6 por tratar-se apenas de um fator de normalização. Dessa forma, pode-se classificar o documento de acordo com a sua classe mais provável através da Equação 2.7:

$$c = argmax_C \ P(C = c_i|D = d) \tag{2.7}$$

,

onde o operador $argmax$ busca o parâmetro C que maximiza $P(C = c_i|D = d)$.

O modelo Naïve Bayes baseado nas Equações 2.7 e 2.6 assume que as variáveis aleatórias W_i possuem uma distribuição de Bernoulli, ou seja, assumem somente o valor 0 ou 1. Esse modelo, chamado de Bernoulli Naïve Bayes, leva em conta somente a presença ou ausência de cada palavra. Esse método é adequado quando os documentos são curtos (frases, *tweets* e etc) e o vocabulário é pequeno. Entretanto, esse tipo de informação sobre a palavra é fracamente relacionada com a classe do documento, o que pode produzir ruído no momento da avaliação do classificador. Documentos mais longos (parágrafos, notícias completas e etc...) tendem a possuir palavras repetidas e esta abordagem ignora as frequências das palavras. Para contornar essa questão, modelo Multinomial será utilizado neste trabalho.

No modelo multinomial, os termos dos documentos são tratados amostras de uma distribuição multinomial. Dessa forma calculamos a probabilidade a posteriori $P(C = c|D = d)$ de acordo com a equação 2.8, onde w_i é a frequência da palavra W_i no documento. Da mesma forma como no modelo de Bernoulli, a classificação do documento de acordo com sua classe mais provável é realizada pela Equação 2.7.

$$P(C = c|D = d) =\propto P(C = c)(\Pi_{i=0}^{m} P(W_i = w_i, C = c)^{w_i}) \tag{2.8}$$

Muitas vezes, pode ocorrer que a frequência de uma determinada palavra ser 0 e, desse modo anulando-se o valor da probabilidade posterior tanto do modelo de Bernoulli como do Multinomial. Uma estratégia utilizada para evitar esse tipo de fenômeno é adicionar 1 a todas as frequências dos termos presentes. Esse técnica é conhecida como *add-one smothing*, e é utilizada pelo fato de não alterar as proporções entre as frequências de cada termo.

2.2.2 Regressão Logística Multinomial (Modelo de Máxima Entropia)

Os classificadores generativos calculam a probabilidade $P(x|y)$ a partir da probabilidade a priori $P(y)$. O classificador Naïve Bayes é um exemplo de modelo generativo. Já os classificadores discriminantes estimam a probabilidade $P(x|y)$ diretamente, sem a informação a priori $P(y)$. Os classificadores SVM e Regressão Logística são bons exemplos desse tipo de modelo.

O classificador de Regressão Logística Multinomial é também denominado por Modelagem de Máxima Entropia no campo de processamento de linguagem Natural. Tecnicamente, a Regressão Logística é utilizada para classificar as entradas em apenas uma de duas classes possíveis, já a Regressão Logística Multinomial é utilizada para classificar uma entrada quando existem mais de duas classes possíveis. Neste trabalho, vamos nos referir a Regressão Logística Multinomial apenas por Regressão Logística ou MaxEnt, para abreviar a nomenclatura. Este classificador pode ser uma alternativa ao modelo

Naïve Bayes, já que, ao contrário deste último, consegue lidar com dados fortemente correlacionados (BERGER; PIETRA; PIETRA, 1996).

O modelo utilizado de Regressão Logística consiste em uma soma ponderada por w_i das características extraídas f_i do texto para cada classe:

$$p(y = c|\mathbf{x}) = \sum_i w_i f_i(\mathbf{x}, c) = \mathbf{w}^T \mathbf{f} \tag{2.9}$$

Entretanto, a Equação 2.9 não pode ser usada para calcular $P(y|x)$ porque o resultado dessa equação pode variar de $-\infty$ a ∞ o que faz com que a Equação 2.9 não seja uma probabilidade. Para limitar essa soma somente para valores positivos, podemos usar o somatório como expoente de uma função exponencial normalizada por um fator Z para limitar seus valores ao intervalo [0,1].

$$p(y = C|\mathbf{x}) = \frac{1}{Z} e^{\sum_i w_i f_i(\mathbf{x},c)} \tag{2.10}$$

$$p(y = C|\mathbf{x}) = \frac{1}{\sum_c e^{\sum_i w_i f_i(\mathbf{x},c)}} \exp \sum_i w_i f_i(\mathbf{x}, \mathbf{c}) \tag{2.11}$$

A variável f_i é uma função binária que indica a presença de uma característica (atributo) na entrada. Dessa forma, f_i é uma função da variável de entrada, neste contexto, o texto em sua representação vetorial *bag-of-words*. Entretanto, f_i também deve levar em conta a classe a qual o documento pertence. Dessa forma, f_i torna-se uma função tanto do documento ($\mathbf{x}$) como da classe (c),

$$f_i(\mathbf{x}, c) = \begin{cases} 1, \text{ Se "ótimo"} \subset \mathbf{x} \text{ e } c = \text{positivo} \\ 0, \text{ caso contrário.} \end{cases} \tag{2.12}$$

A probabilidade calculada na Equação 2.11 é diretamente proporcional ao expoente do numerador da fração. Sendo o denominador constante para todos os valores de $\mathbf{x}$, a Equação 2.11 pode reduzida para a seguinte expressão,

$$\hat{c} = \text{argmax}_c \sum_i w_i f_i(\mathbf{x}, c). \tag{2.13}$$

Portanto, a classe $\hat{c}$ que maximiza a Equação 2.13 é a classe mais provável do documento representado por $\mathbf{x}$.

Dessa forma, é necessário encontrar os parâmetros ótimos w_i. Para um dado documento $\mathbf{x}^{(j)}$ cuja classe é $y^{(j)}$, o conjunto de parâmetros w_i é dado pelo conjunto que satisfaz a Equação 2.15.

$$\hat{w} = \text{argmax}_w P(y^{(j)}|x^{(j)}), \tag{2.14}$$

$$L(w) = \sum_j \log P(y^{(j)}|x^{(j)}). \tag{2.15}$$

A Equação 2.15 é chamada de função de verosimilhança e constitui a função objetivo que será maximizada. Substituindo a Equação 2.11 na Equação 2.15 e utilizando a notação vetorial para $\mathbf{w} = [w_1...w_n]$ e $\mathbf{F}(\mathbf{x}, c) = [f_1(\mathbf{x}, c)...f_n(\mathbf{x}, c)]$, temos,

$$L(w) = \sum_j \log \frac{e^{\mathbf{wF}(\mathbf{x}^{(j)}, y^{(j)})}}{\sum_y e^{\mathbf{wF}(\mathbf{x}^{(j)}, y^{(j)})}}, \tag{2.16}$$

$$L(w) = \sum_j [\log(e^{\mathbf{xF}(\mathbf{x}^{(j)}, y^{(j)})} - \log \sum_{y' \subset Y} (e^{\mathbf{xF}(\mathbf{x}^{(j)}, y^{(j)})}), \tag{2.17}$$

$$L(w) = \sum_j \mathbf{xF}(\mathbf{x}^{(j)}, y^{(j)}) - \sum_j \log \sum_{y' \subset Y} (e^{\mathbf{xF}(\mathbf{x}^{(j)}, y^{(j)})}). \tag{2.18}$$

A função de máxima verossimilhança expandida apresentada na Equação 2.18 constitui a função objetivo a ser maximizada, que vem a ser um problema de otimização convexa. Para encontrar os parâmetros que maximizam essa função, derivamos $L(w)$ em relação à w_i,

$$\frac{\partial L(w)}{\partial w_k} = \sum_j f_k(y_{(j)}, \mathbf{x}^{(j)}) - \sum_j \sum_y P(y, \mathbf{x}^{(j)}) f_k(y^{(j)}, \mathbf{x}^{(j)}), \tag{2.19}$$

$$\frac{\partial L(w)}{\partial w_k} = \sum_j f_k(y_{(j)}, \mathbf{x}^{(j)}) - \sum_j \sum_y \frac{e^{\mathbf{wF}(\mathbf{x}^{(j)}, y^{(j)})}}{\sum_y e^{\mathbf{wF}(\mathbf{x}^{(j)}, y^{(j)})}} f_k(\mathbf{x}^{(j)}, y^{(j)}). \tag{2.20}$$

O primeiro termo do membro direito da Equação 2.20 trata-se do número de ocorrências da característica indicada pela função f_k na coleção de documentos. O segundo termo, o único que depende de $\mathbf{w}$, é o valor esperado de f_k. Ao igualarmos a Equação 2.20 à zero chegamos à conclusão que os valores ótimos de w_k são aqueles igualam o valor esperado de f_k ao número de ocorrências do mesmo, sendo

$$\frac{\partial L(w)}{\partial w_k} = \text{Num. Ocorrências}(f_k) - E[f_k] = 0,$$
$$\text{Num. Ocorrências}(f_k) = E[f_k].$$

2.2.3 Máquina de Vetor de suporte(*Support Vector Machine* - SVM)

A Máquina de Vetor de Suporte (*Support Vector Machine* - SVM) é um método de classificação baseado na teoria da Otimização Convexa. A SVM procura determinar o

melhor hiperplano de separação que pode haver entre dois conjuntos (classes) de dados linearmente separáveis (HAYKIN et al., 2009).

Dado um conjunto de dados linearmente separável, há uma infinidade de hiperplanos que podem servir como classificadores (Figura 2). A SVM busca pelo hiperplano ótimo, isto é, aquele que possui a maior margem de distância para as duas classes. Esta margem é definida como a distância entre o ponto mais próximo e o hiperplano de separação. Portanto, o hiperplano deve estar o mais afastado possível tanto do conjunto de um dados como do outro.

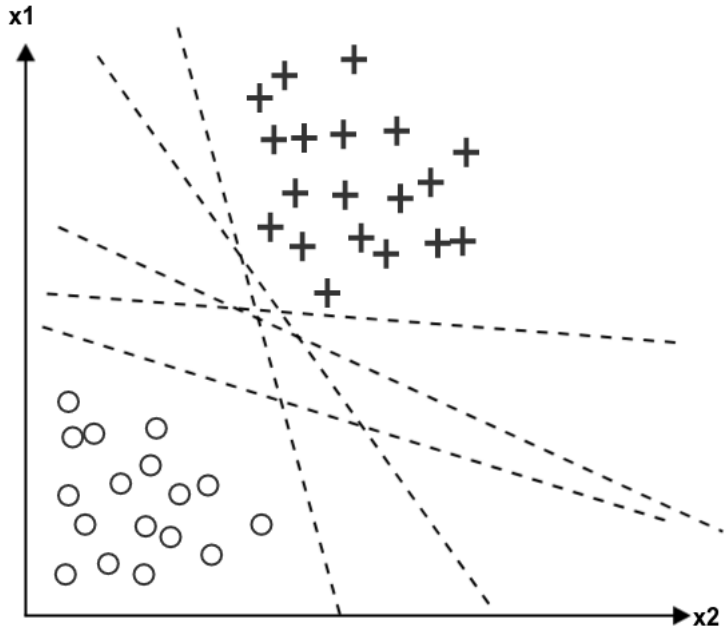

Figura 2 – Exemplo da infinidade de separadores lineares que pode haver entre dois conjuntos de dados linearmente separável

Considerando um conjunto de treinamento $\{\mathbf{x_i}, d_i\}$, onde $\mathbf{x_i}$ é o vetor de entrada e a d_i é saída desejada (rótulo da classe) do processo de classificação. Em um problema de classificação com duas classes, d_i pode assumir os valores 1 ou -1. A equação do hiperplano separador é dada pela Equação 2.21:

$$\mathbf{w}^T\mathbf{x} + b = 0, \tag{2.21}$$

onde o vetor $\mathbf{w}$ é um vetor ajustável e b é o coeficiente linear do hiperplano. Para o caso de classificação com duas classes, podemos escrever a função $g(\mathbf{x})$ de modo que:

$$g(\mathbf{x_i}) = \mathbf{w}^T\mathbf{x_i} + b \geq 0 \text{ para } d_i = +1 \tag{2.22}$$

$$g(\mathbf{x_i}) = \mathbf{w}^T\mathbf{x_i} + b < 0 \text{ para } d_i = -1. \tag{2.23}$$

A tarefa da SVM é encontrar os valores ótimos de $\mathbf{w}$ e b de modo que a margem seja a máxima (Figura 3). Afim de calcular a distancia de cada vetor $\mathbf{x}$ ao hiperplano, podemos representá-los através da expressão na Equação 2.24:

$$\mathbf{x} = \mathbf{x}' + r\frac{\mathbf{w}}{||\mathbf{w}||}, \tag{2.24}$$

onde $\mathbf{x}'$ é a projeção ortogonal do vetor $\mathbf{x}$ no hiperplano ótimo e r é a distância euclidiana entre a projeção $\mathbf{x}'$ e o vetor $\mathbf{x}$.

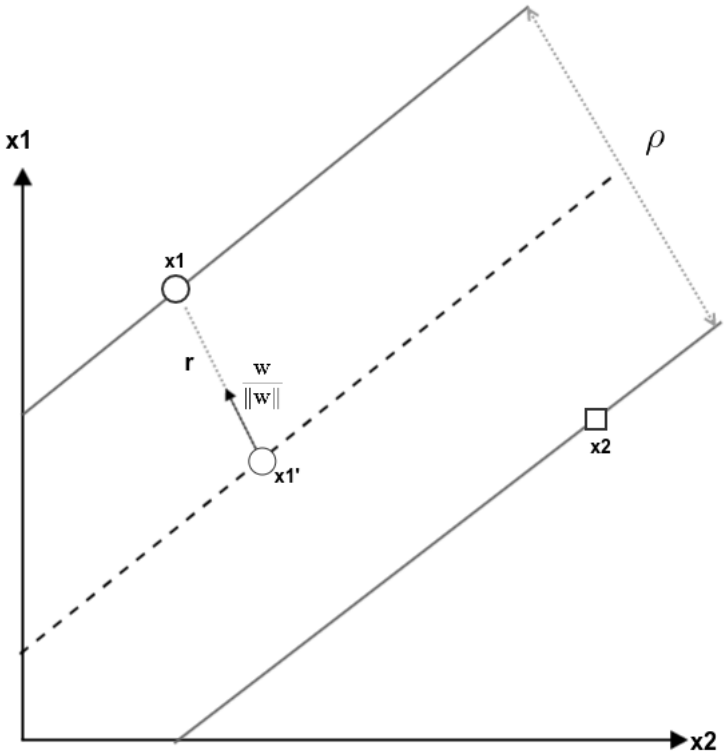

Figura 3 – Ilustração da ideia do hiperplano ótimo (linha tracejada) com margem máxima (linhas cheias).

Os vetores que satisfazem com igualdade alguma das duas condições da Equação 2.23 são chamados de vetores de suporte. Os outros vetores que não são de suporte são irrelevantes para o cálculo da margem.

A margem é dada pela distância dos vetores suporte até o hiperplano. Essa distância pode ser calculada através da substituição da Equação 2.24 na Equação 2.23, de onde podemos derivar os seguintes valores para a distância r, como,

$$r = \frac{+1}{||\mathbf{w}||} \text{ para } d_i = +1 \tag{2.25}$$

$$r = \frac{-1}{||\mathbf{w}||} \text{ para } d_i = -1. \tag{2.26}$$

Dessa forma definimos matematicamente a margem de separação como

$$\rho = \frac{2}{||\mathbf{w}||}. \tag{2.27}$$

Portanto, maximizar a margem de separação ρ significa minimizar a norma euclidiana do vetor $\mathbf{w}$. Para isso, estabelecemos 4 passos básicos na formulação do problema da SVM:

1. Formular o problema de otimização convexa no espaço dos vetores $< \mathbf{w}$;
2. Construir a função Lagrangiana;
3. Encontrar as condições de optimalidade;
4. Resolver o problema de otimização no espaço dual dos multiplicadores de Lagrange.

Então, dado o conjunto de treinamento $\{\mathbf{x_i}, d_i\}$ e sabendo que $g(\mathbf{x}_i \geq 1$ ou $g(\mathbf{x}_i < 1$ dependendo de d_i podemos formular o problema de otimização convexa (problema primal) como:

$$\text{Minimizar } \Phi(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} \tag{2.28}$$

$$\text{sujeito à restrição: } d_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1. \tag{2.29}$$

A partir daí, utilizamos os multiplicadores de Lagrange α para construir a função Langrangiana (Equação 2.30):

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{N}[d_i(\mathbf{w}^T\mathbf{x}_i + b) - 1]\alpha_i. \tag{2.30}$$

Com a função Langrangiana em mãos, devemos minimizá-la em relação à $\mathbf{w}$ e b, resultando nas Equações 2.31 e 2.32:

$$\frac{\partial J}{\partial \mathbf{w}} = 0 \longrightarrow \mathbf{w} = \sum_{i=1}^{N}\alpha_i d_i \mathbf{x}_i \tag{2.31}$$

$$\frac{\partial J}{\partial b} = 0 \longrightarrow \sum_{i=1}^{N}\alpha_i d_i = 0. \tag{2.32}$$

A partir daí podemos formular outro problema de otimização, o problema dual, onde a solução fornece os valores ótimo de α_i, que por sua vez, fornecem os valores ótimos de $\mathbf{w}$ e b.

Expandindo a função J (Equação 2.30), temos:

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{N} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b\sum_{i=1}^{N} \alpha_i d_i + \sum_{i=1}^{N} \alpha_i. \tag{2.33}$$

Substituindo os resultados encontrados no lado direito das Equações 2.31 e 2.32, encontramos a Equação 2.34, que é uma função unicamente dos multiplicadores de Lagrange α:

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\sum_{i}\sum_{j} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^{N} \alpha_i = Q(\alpha). \tag{2.34}$$

O problema dual consiste em maximizar a função $Q(\alpha)$ em relação a α. A solução ótima do problema dual também fornece a solução ótima do problema primal. Nosso objetivo agora é encontrar os valores α_i que maximizam $Q(\alpha)$ sujeito as condições:

$$\sum_{i=0}^{N} \alpha_i d_i = 0,$$
$$\alpha_i \geq 0.$$

Diferentemente do problema de otimização primal, o problema dual não precisa ser resolvido através do Lagrangiano, dependendo somente dos vetores do conjunto de treinamento na forma de produtos internos $\mathbf{x}_i^T \mathbf{x}_j$.

2.3 Métodos de Seleção de Atributos

Afim de evitar o aumento excessivo de dimensões (, estouro de dimensões e a consequente esparsidade da representação *bag-of-words*, reduzimos a quantidade de atributos (*ngrams* existentes) a um conjunto menor, porém mais significativo, de acordo com um determinado critério. Essa metodologia é chamada de seleção de atributos.

Os métodos utilizados para selecionar atributos neste trabalho foram o Teste Qui Quadrado, Categorical Proportion Difference (CPD) e o Categorical Probability Proportional Difference (CPPD). O método Qui Quadrado foi escolhido em vista de sua ampla utilização em problemas de classificação de texto (YANG et al., 2012; HADDI; LIU; SHI, 2013; SHARMA; DEY, 2012), bem como os outros dois últimos que são métodos para tratamento de problemas de Categorização de Texto e, ocasionalmente, Análise de Sentimento.

2.3.1 Teste Qui Quadrado

Este método mede o nível de dependência estatística entre duas variáveis aleatórias: a distribuição de um determinado *ngram* ao logo dos textos e as suas respectivas classes. Para calcular o valor qui quadrado primeiro é necessário construir uma tabela de contingência 2x2 para cada *ngram*, semelhante à Tabela 3

Tabela 3 – Tabela de contingência para o cálculo do valor qui quadrado. **w** indica a ocorrência do ngram num documento e **w** indica a ausência. **c** indica que o documento pertence a classe c enquanto **c** indica o contrário.

	c	**∼c**
w	A	B
∼w	C	D

Os valores contidos na tabela são estatísticas do ngram dentro do *corpus*. A é quantidade de documentos da classe c que contém o *ngram* w. B indica o número de documentos fora da classe c que contém o *ngram* w. De forma semelhante às contagens A e B, os valores C e D contam o mesmo evento, mas para a ausência do *ngram* w. De posse desses valores, podemos calcular o valor qui quadrado de um *ngram* w para a classe c através da Equação 2.35 (N é o numero de documentos):

$$\chi^2(w,c) = \frac{N \times (AC - BD)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{2.35}$$

O valor qui quadrado assume o valor 0 quando as duas variáveis são estatisticamente independentes e cresce a medida que as variáveis tornam dependentes uma da outra. Esse valor é utilizado para selecionar os *ngrams* que estão mais relacionados com uma determinada classe. Quanto maior o valor qui quadrado mais relevante é esse *ngram* para aquela classe.

2.3.2 Diferença Proporcional Categórica

O método da Diferença Proporcional Categórica (Categorical Proportional Difference - CPD), proposto em Simeon e Hilderman (2008), foi originalmente proposto para tarefas de categorização de texto, uma aplicação específica de classificação de texto. Trata-se de técnica supervisionada que atribui para cada palavra um peso de importância para cada categoria de interesse da classificação. O pressuposto utilizado primariamente é de que a frequência de palavras específicas de um certo assunto caracteriza o tópico majoritário daquele texto. Por exemplo, as palavras "artilheiro", "goleada"e "brasileirão"relacionam-se ao assunto Futebol, por isso textos que contenham essas palavras são altamente prováveis de tratarem do assunto futebol. Entretanto, as palavra "artilheiro"também relaciona-se

com o assunto Guerra juntamente com as palavras "soldado"e "tropa". Portanto somente a ocorrência da palavra artilheiro não é suficiente para a determinação do assunto do referido texto. O método CPD funciona da seguinte forma: para um dado *corpus* rotulado, extraímos todas palavras que ocorrem nele (com exceção de *stopwords*) e construímos a seguinte tabela de contingência (Tabela 4):

Tabela 4 − Tabela de contigência

Palavras	Categorias	
	c	c
w	A	B
∼w	C	D

Onde A é o numero de vezes que a palavra w ocorre em documentos da categoria c, B é o numero de vezes que a palavra w ocorre em documentos fora da categoria c, C é o numero de documentos em que a palavra w não ocorre em documentos da categoria c, D é o numero de documentos em que a palavra w não ocorre em documentos fora da categoria c. De posse dessa informação, o valor CPD da palavra w para a classe c é calculado de acordo com a Equação 2.36:

$$CPD(w, c) = \frac{A - B}{A + B} \tag{2.36}$$

O valor CPD de uma palavra para uma categoria mede o quanto uma palavra contribui para diferenciar uma categoria das outras dentro do mesmo corpus. Se ela ocorre muito em documentos de uma determinada categoria, ela é um forte indicador desta classe. Se ela ocorre de modo igualmente disperso entre as categorias de textos, essa palavra não carrega muita informação para distinção das categorias existentes.

O valor calculado é compreendido no intervalo [-1, 1], onde 1 indica que aquela palavra ocorre somente em documentos daquela categoria e -1 indica que a aquela palavra ocorre em todas as outras categorias menos naquela categoria de interesse. O valor CPD de uma palavra é o valor CPD da palavra para a categoria cujo valor é máximo (Equação 2.37):

$$CPD(w) = max_i\{CPD(w, c_i)\} \tag{2.37}$$

O valor CPD pode ser utilizado em aplicações de Análise de Sentimento como uma ferramenta de seleção de atributos. A papel da técnica é selecionar um conjunto de *ngrams* que melhor caracterize as classes de interesse (positivo, negativo e neutro) afim reduzir a quantidade de palavras a serem consideradas no modelo *bag-of-words*. Dentro de

um *corpus* rotulado para as três classes, calculamos o valor CPD para todas as palavras que compõe a partir de uma tabela de contingência semelhante à Tabela 4:

Tabela 5 – Exemplo de tabela de contingência já com cálculo do CPD

ngrams	**Classes**			
	positivo	**negativo**	**neutro**	**CPD**
"absolutamente vergonhoso"	4	35	1	0,75
"acabar piorando"	2	49	3	0,81
"ótimo bom"	63	9	6	0.63
"afirmou"	20	26	22	-0.23

Perceba que a palavra "afirmou"possui um valor CPD muito baixo (-0,23) pelo fato de ela estar quase que uniformemente distribuída entre as classes do problema. Como ela é bastante frequente em todas classes, ela não caracteriza adequadamente nenhuma delas. Feito o calculo, é estabelecido empiricamente um valor de CPD mínimo para admitir um ngram como relevante. A partir daí esses ngrams selecionados serão os únicos a serem contado no modelo bag-of-words.

2.3.3 Diferença Proporcional Categórica Probabilistica

O método da Diferença Proporcional Categórica Probabilistica (Categorical Probability Proportion Difference - CPPD) proposto por Agarwal e Mittal (2012) trata-se de uma melhoria do método anterior. Além de calcular o valor CPD de cada *ngram*, o CPPD ordena os *ngrams* por sua respectiva probabilidade de ocorrência dentro da classe. Enquanto o CPD calcula o quanto um determinado *ngram* pertence a uma classe, o CPPD mede o quanto ele é importante dentro de uma classe. Por exemplo: duas palavras distintas como "excelente"e "antológico"podem ter o valor CPD igual a 1 na classe positiva, porém se a probabilidade de ocorrência de "excelente"for bem maior do que a de "antológico", significa que a classe positiva é melhor representada pela palavra "excelente"do que "antológico".

O calculo da probabilidade de ocorrência de cada ngram é dado por

$$p(w) = \frac{C(w)}{\sum_i^N C(w_i)}, \tag{2.38}$$

onde $C(w)$ é a contagem das ocorrências do ngram w e N é o numero de total de palavras daquela classe. O somatório no denominador da Equação 2.38 é a contagem total das ocorrências de todas as palavras daquela classe. Com os *ngrams* ordenados por sua respectiva probabilidade de ocorrência, o critério de escolha dos ngrams passa a ser tanto o valor CPD como a probabilidade de ocorrência.

3 Materiais e Métodos

Neste capítulo é descrita a abordagem para avaliar e comparar os métodos de seleção de atributos e os classificadores utilizados. Dado o parágrafo de um texto jornalístico, pretende-se classificá-lo em uma destas três classes: positivo, negativo ou neutro.

A Figura 4 ilustra num diagrama de blocos a metodologia utilizada para conduzir os experimentos deste trabalho Numa sequência de passos que divide em dois caminhos, cada qual representando um experimento.

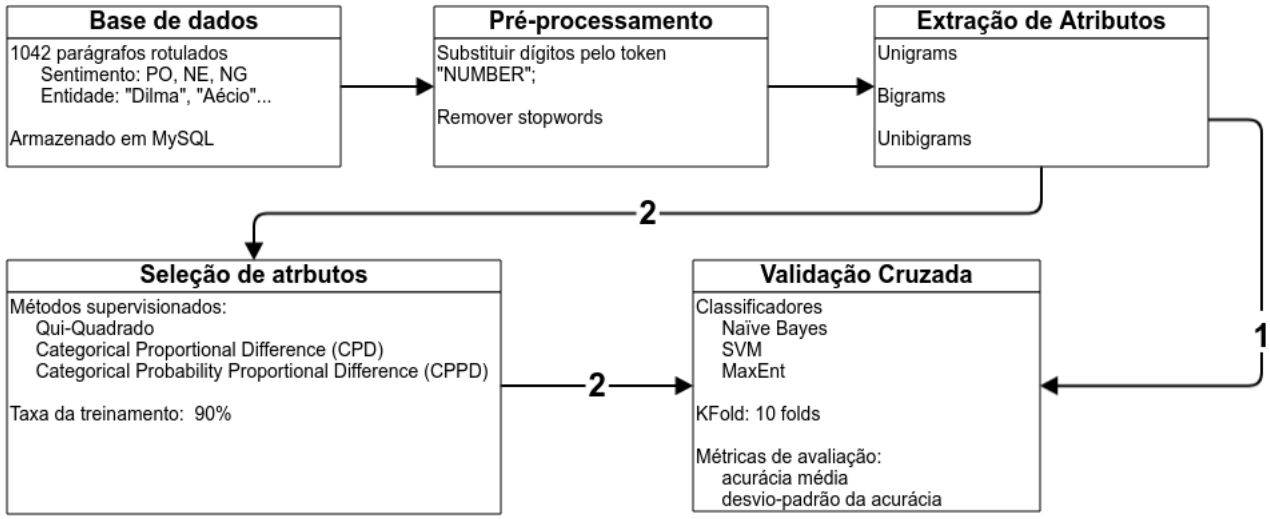

Figura 4 – Metodologia ilustrada num diagrama sequencial

A princípio foi obtida uma base de dados já rotulada de notícias relacionadas as eleições, em seguida algumas etapas de preprocessamento foram aplicadas aos textos. A partir daí é possível extrair os atributos de cada texto utilizando representações *bag-of-words* com modelos diferentes, sendo cada texto representado por 3 modelos: *unigram*, *bigram* e *unibigram*. Feito isso, dois experimentos são realizados. O primeiro consiste em aplicar os classificadores diretamente nas representações do texto e, através das métricas de desempenhos estabelecidas neste capítulo, verificar qual modelo é mais adequado para a tarefa de classificação. No segundo experimento, o modelo escolhido na etapa passada é submetido aos métodos de seleção de atributos afim verificar como cada um desses impacta o desempenho do classificador na distinção dos textos.

Os experimentos foram conduzidos utilizando a linguagem de programação Python juntamente com as bibliotecas Scikit Learn [1], para as aplicações de aprendizagem de máquina, e Natural Language Toolkit (NLTK) [2], que possui ferramentas de manipulação

[1] http://scikit-learn.org/
[2] http://www.nltk.org/

de texto.

3.1 Base de Dados - *Corpus* Viés

A base de dados utilizada neste trabalho, o Corpus Viés, foi coletada e rotulada por Arruda, Roman e Monteiro (2015). Corpus Viés consiste numa coleção de 131 artigos de notícias on-line sobre as eleições de 2014 para o governador de São Paulo e para o presidente do Brasil. Estes artigos foram obtidos de cinco fontes bem conhecidas no Brasil: Veja; Estadão; Folha; G1 e Carta Capital. Os textos que compõem os artigos são divididos em parágrafos, sendo que, cada um foi rotulado manualmente por quatro anotadores em relação à orientação de sentimento apresentada no texto, positivo ("PO"), negativo ("NE") ou neutro ("NE"). Esta base de dados totaliza 1042 parágrafos, 310 rotulados como positivos, 391 negativos e 341 neutros.

Este *corpus* foi construído coletando-se as notícias dos perfis públicos dos referidos noticiários no *Twitter* no período de 06/09/2014 até 12/09/2014. Cada paragrafo foi rotulado manualmente por 4 pessoas. Estes determinaram a tanto a polaridade de cada parágrafo quanto a entidade a qual esse se referia. O *Corpus* Viés é bem balanceado em relação aos seus rótulos, possuindo proporções equilibradas entre as classes, porém em relação aos noticiários, esta base de dados é fortemente desbalanceada dada a baixa atividade de alguns perfis no *Twitter*, conforme a figura 5.

O Corpus Viés é originalmente armazenado no formato XML mas para este trabalho ele foi convertido convertido e armazenado em um banco de dados MySQL para facilitar o acesso e consultas aos textos e seus rótulos. A base de dados foi pré-processada removendo *stopwords*, convertendo todos os caracteres para minúsculo e também substituindo todos os algarismos presentes por um token "NUMBER".

3.2 Extração dos Atributos do Texto

Cada parágrafo foi representado de acordo com o modelo *bag-of-words*, conforme já apresentado no Capítulo 2. No primeiro experimento, 3 modelos foram utilizados: *unigram*, *bigram* e *unigram* e, *bigram* simultâneamente, que por conveniência será chamado de *unibigram*. Nessa etapa, todas as palavras encontradas no *corpus* pré-processado compõem o vocabulário. A dimensão do vetor que representa cada texto é aproximadamente n, n^2 e $n(n+1)$ respectivamente para os modelos *unigram*, *bigram* e *unibigram*, onde n é o tamanho do vocabulário. A tabela 6 exibe as respectivas dimensões para o *Corpus* Viés.

Afim de diminuir a dimensionalidade da representação vetorial do texto, empregamos os métodos de seleção de atributos. Essas técnicas consistem em encontrar um subconjunto de atributos (*ngram* em nosso caso) que contenha mais informação acerca

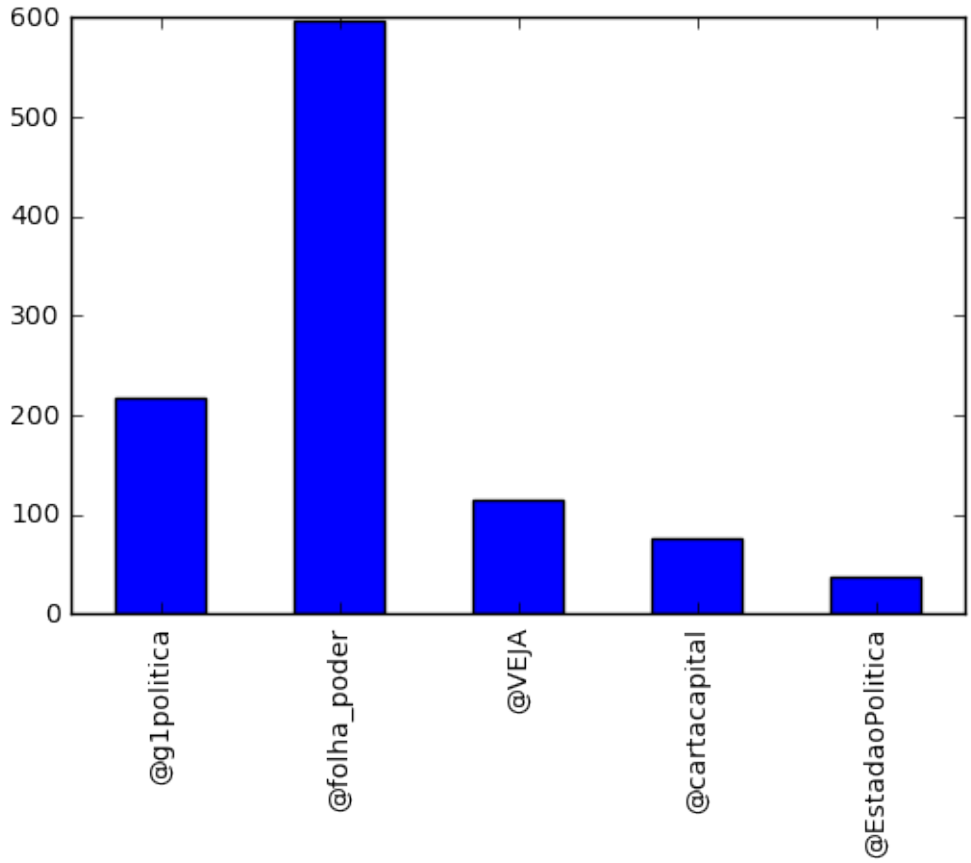

Figura 5 – Distribuição dos parágrafos em função dos noticiários

Modelo	**Dimensionalidade**
unigram	5344
bigram	18958
unibigram	24302

Tabela 6 – Dimensionalidade do vetor de representação do texto para modelo *bag-of-words* utilizado

das classes de interesse do problema. No segundo experimento aplicamos essas técnicas afim de identificar, através da performance dos classificadores, qual método encontra o melhor conjunto de atributos para descrever cada classe.

3.3 Treinamento dos Métodos de Seleção de Atributos e dos Classificadores

Para cada modelo *bag-of-words*, três classificadores foram treinados com as mesmas parametrizações: Naive Bayes Mutinomial com *add-one smoothing*, Regressão logística com um modelo de Máxima Entropia e uma *Support Vector Machine* usando como *kernel* uma rede neural de base radial com um parâmetro C ajustado empiricamente no valor

316, este ultimo.

Por serem métodos supervisionados, as técnicas de seleção de atributos usadas aqui (Qui Quadrado, CPD e CPPD) necessitam passar por uma etapa de treinamento com uma parcela do *corpus*. Para todos os três métodos foi utilizado 90% dos parágrafos contidos no *corpus*. No fim deste treinamento, cada método irá gerar um conjunto de *ngrams* de tamanho igual ao número de variáveis selecionadas. O vocabulário do modelo *bag-of-words* será composto por esse conjunto de *ngrams* selecionados.

No método Qui Quadrado, as variáveis com o maior valor qui quadrado foram selecionadas. Para os métodos CPD e CPPD foi utilizado um valor de corte para o valor CPD igual a 1. Dessa forma, somente palavras que aparecem exclusivamente em uma das três classes irão compor o vocabulário. Para os três métodos, o número de atributos selecionados foi variado discretamente num intervalo de 1000 a 5000 termos com um passo de 1000.

3.4 Metodologia e Métrica de Avaliação

A metodologia empregada para avaliar os resultados é composta por duas etapas: Validação Cruzada (*Cross Validation*) *K-Fold* e o cálculo da acurácia média.

3.4.1 Acurácia

Em um problema de classificação, podemos descrever os resultados através de uma tabela semelhante a Tabela 7. Essa tabela também é conhecida por matriz de confusão.

Tabela 7 – Exemplo de matriz de confusão

		Predição	
		A	**¬A**
Real	**A**	VP	FN
	¬A	FP	VN

A Tabela 7 descreve um problema de classificação onde uma instância dos dados pode ser classificada como pertencendo a classe **A** ou não pertencendo a esta classe, ¬ **A**. Nas linhas dessa tabela temos a informação de quantos elementos pertencem a classe **A** e quantos não pertencem. Nas colunas, temos a informação de quantos elementos foram classificados como pertencendo a classe **A** e quantos classificados como não pertencentes.

As siglas adotadas na Tabela 7, indicam as quantidades de classificações realizadas corretamente e erroneamente para cada classe, conforme a seguir:

- Verdadeiros Positivos (**VP**): Quantidade de instâncias da classe **A** que foram classificadas como pertencendo a **A**;

- Falsos Positivos (**FP**): Quantidade de instâncias fora da classe **A** que foram classificadas como pertencendo a **A**;
- Verdadeiros Negativos (**VN**): Quantidade de instâncias fora da classe **A** que foram classificadas como não pertencendo a **A**;
- Falsos Negativos (**FN**): Quantidade de instâncias da classe **A** que foram classificadas como não pertencendo a **A**.

A partir dessa tabela, definimos a acurácia como uma métrica de desempenho geral do classificador (Equação 3.1). A acurácia mede, em proporção, quantas instâncias foram classificadas corretamente em suas respectivas classes.

$$\text{Acurácia} = \frac{VP + VN}{VP + VN + FP + FN} \tag{3.1}$$

Outras métricas como sensibilidade e especificidade não foram adotadas neste trabalho por serem dedicadas a avaliar base de dados desbalanceadas. No caso abordado neste trabalho, o desbalanceamento do corpus é insignificante e a acurácia pode ser utilizada como uma única métrica de avaliação para todas as classes.

3.5 Validação Cruzada *K-Fold*

Ao avaliar um classificador, geralmente divide-se os dados utilizados em dois conjuntos: um para treinamento e ajuste dos parâmetros e outro para teste, onde a performance do classificador é medida. Pelo fato do conjunto de teste ser, na maioria das vezes, menor que o conjunto de treinamento, há o risco deste não ser suficientemente representativo para todo o conjunto, e portanto, a avaliação feita ser ruidosa.

Afim de evitar esse tipo de erro de avaliação, a estratégia de validação cruzada *K-Fold* foi utilizada. Essa técnica encontra-se ilustrada na Figura 6. O conjunto de dados é dividido em K subconjuntos distintos separando $K-1$ subconjuntos para treinamento e o restante para teste. Esse procedimento de divisão é realizado K vezes, criando-se assim K combinações de subconjuntos de treinamento e de teste, de forma que cada um dos subconjuntos compõem o conjunto de teste em cada combinação. Cada combinação de subconjuntos de treinamento e teste é chamada de *fold*, daí o nome *K-Fold*.

Em cada *fold* são calculadas as métricas de desempenho, em nosso caso, a acurácia. Afim de obter-se uma perspectiva geral do desempenho sobre todos os *folds*, a média e o desvio padrão da acurácia foram considerados como métricas de desempenho. Dessa forma estabelecemos que um bom classificador é aquele que possui uma alta acurácia média e um baixo desvio padrão.

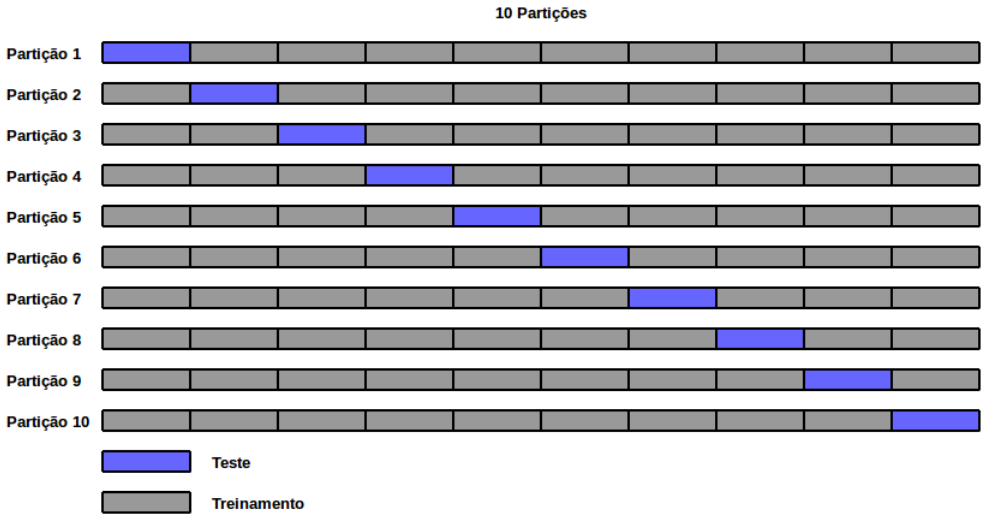

Figura 6 – Exemplo de validação cruzada *K-Fold* utilizando 10 *Folds*

4 Resultados e Discussão

Neste capítulo apresentaremos os resultados obtidos dos experimentos descritos no capitulo anterior. No primeiro experimento, comparamos 3 modelos de representação textual e no segundo comparamos o impacto dos métodos de seleção de atributos sobre o modelo de representação com a melhor performance no experimento anterior. Os resultados do primeiro experimento justificarão o uso da representação vetorial escolhida para o segundo experimento.

4.1 Primeiro Experimento - Classificação sem Seleção Atributos

Os resultados da avaliação de desempenho dos algoritmos Naive Bayes, SVM e Regressão Logística para classificar os texto sem seleção de atributos estão descritos na Tabela 8.

Tabela 8 – Resultados da classificação sem seleção de atributos

Modelo	**Dimensionalidade**	**Acurácia**		
		Naïve Bayes	**SVM**	**Regressão Logística**
Unigram	5344	59.1	55.16	56.12
Bigram	18958	55.46	52.77	50.95
Unibigram	**24302**	**59.3**	**57.18**	**56.99**

Características do tipo *unigram*, *bigram* e *unibigram* apresentaram acurácias parecidas, porém melhores desempenhos foram proporcionadas para atributos do tipo *unibigram* com de 59,3% (Naive Bayes), 57,18% (SVM) e 56,99% (Regressão Logística / MaxEnt). Observamos que não há melhorias ao usar bigrams em vez de unigrams. A partir desta experiência, vemos que um vetor de características composto por *unigramas* e *bigram* (*unibigram*) é a melhor representação para os parágrafos de notícias. Esta combinação de características afeta positivamente todos os classificadores, mas a um custo de alta dimensionalidade. Além disso, o classificador Naive Bayes obteve melhor acuraria em todas as atributos selecionadas: *unigram* (59.1%), *bigram* (55,46%) e *unbigram* (59,3%). Isso pode sugerir a independência estatística entre os atributos do texto (*ngrams*).

Apesar de possuir mais dimensões que os outros modelos, a representação *unibigram* foi escolhida por apresentar uma performance melhor para todos os classificadores. Mesmo sendo $n + 1$ vezes maior que o modelo *unigram*, o *unibigram* foi escolhido pelo fato de conter informações tanto sobre *unigrams* como *bigrams*. Posteriormente, ao serem aplicados métodos de seleção de atributos, o número de dimensões será reduzido para no mínino $\frac{1}{4}$ do valor atual.

A Tabela 9 apresenta proporcionalmente o quanto a acurácia classificação utilizando *unibigrams* foi aumentada em relação ao uso de *unigrams* e *bigrams*.

Tabela 9 – Comparação proporcional entre a performance dos classificadores com o modelo *unibigram* em relação aos outros.

Modelo	**Naïve Bayes**	**SVM**	**Regressão Logística**
Unigram	0,3%	3,6%	1,55%
Bigram	6,92%	8,35%	11,85%

Embora o modelo com *unibigrams* contemple o uso de *bigrams*, o *unibigram* apresenta uma destacada melhora em todos os classificadores. Já ao comparar-se com o modelo *unigram*, a melhora não é tão proeminente, o que indica que grande parte da informação reside nos unigrams.

4.2 Segundo Experimento - Classificação com Seleção Atributos

No segundo experimento, avaliamos o impacto dos métodos de seleção de atributos, Qui Quadrado, CPD e CPPD, apenas para a combinação de unigrams e bigrams, em um intervalo seleção de 1000 a 5000 atributos.

O impacto do método Qui Quadrado na performance dos classificadores está apresentado na Figura 7. Observamos que o melhor desempenho é alcançado pelo classificador Naïve Bayes, na qual apresenta nível superior de precisão em todos os perfis dos atributos treinados na base de dados em comparação aos demais classificadores. A melhor acuraria alcançada entre os classificadores propostos aparece ao selecionarmos 2000 atributos. Posteriormente, esses valores começam a decair a medida que as atributos são aumentados. Isso indica que alguns atributos importantes para a distinção dos textos não são estatisticamente dependentes dos rótulos atribuídos, e portanto não podem ser encontrados pelo Qui Quadrado. A Tabela 10 apresenta o desvio padrão da acurácia na validação cruzada.

Tabela 10 – Desvios-padrão da acurácia medida em função do classificador e o número de atributos selecionados com a técnica Qui Quadrado

	Num. Atributos				
	1000	**2000**	**3000**	**4000**	**5000**
Naive Bayes	3,3	2,3	2,8	3,4	3,7
MaxEnt	5,0	5,2	4,3	3,7	4,6
SVM	4,9	4,8	3,6	3,7	4,1

O classificador Naïve Bayes é mais preciso com 2000 atributos devido ao seu baixo desvio-padrão em relação aos outros valores do intervalo.

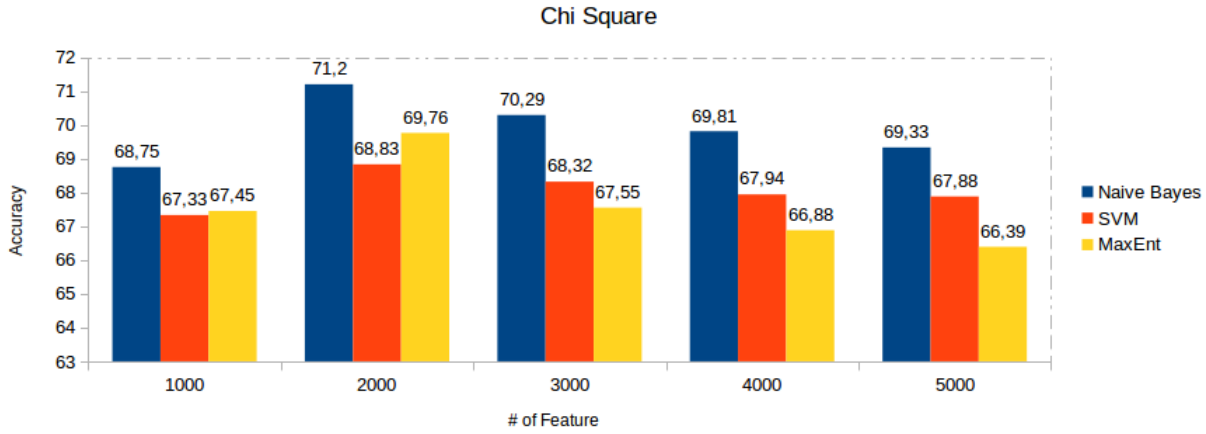

Figura 7 – Performance dos classificadores utilizando o método de seleção Qui Quadrado.

Quando o método CPD é utilizado, observa-se um aumento gradativo do desempenho em relação ao aumento das atributos selecionados (Figura 8), ou seja, a medida que a quantidade de atributos aumenta a performance dos classificadores tende a aumentar. Nesta análise, o algoritmo com melhor desempenho foi Naïve Bayes, que alcançou até 60,75% de acurácia em 5000 atributos. No entanto, apesar do aumento da acurácia relacionada às características, o método CPD obteve fraco desempenho em relação aos demais métodos de seleção avaliados neste trabalho, como Qui Quadrado e CPPD. O método CPD tem uma limitação: embora seja capaz de selecionar *ngrams* exclusivos de uma determinada classe, este método não é capaz de selecionar os melhores *ngrams* dentre os que foram selecionados. A Tabela 10 apresenta o desvio padrão da acurácia na validação cruzada.

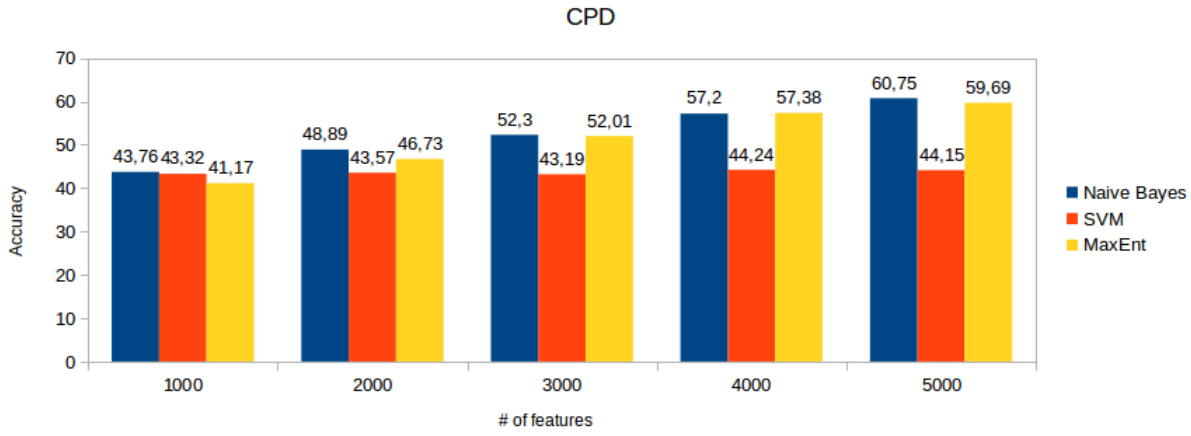

Figura 8 – Performance dos classificadores utilizando o método de seleção CPD.

Neste experimento, o classificador Naïve Bayes é mais preciso com 5000 atributos devido ao seu baixo desvio-padrão em relação aos outros valores do intervalo.

Os melhores resultados foram alcançados utilizando o método CPPD (Figura 9).

Tabela 11 – Desvios-padrão da acurácia medida em função do classificador e o número de atributos selecionados com a técnica CPD

	Num. Atributos				
	1000	**2000**	**3000**	**4000**	**5000**
Naïve Bayes	3,8	3,2	3,6	3,0	2,6
MaxEnt	3,3	3,2	3,2	3,2	3,1
SVM	3,6	3,7	2,2	3,8	4,5

O classificador MaxEnt superou a todos os outros classificadores em todo o intervalo de seleção de atributos em questão. Um fato interessante é baixa variação na performance a partir de 2000 atributos selecionados, o que significa que é possível obter performance equiparável com um modelo mais simples. Em um panorama geral, todos os classificadores obtiveram melhores performances utilizando CPPD como método de seleção em relação aos outros utilizados. Isso se deve ao fato desta técnica cobrir a deficiência encontrada no CPD. O CPPD ordena os ngrams selecionados pelo CPD por sua probabilidade de ocorrência em cada classe, criando assim uma espécie de "ranking" de relevância.

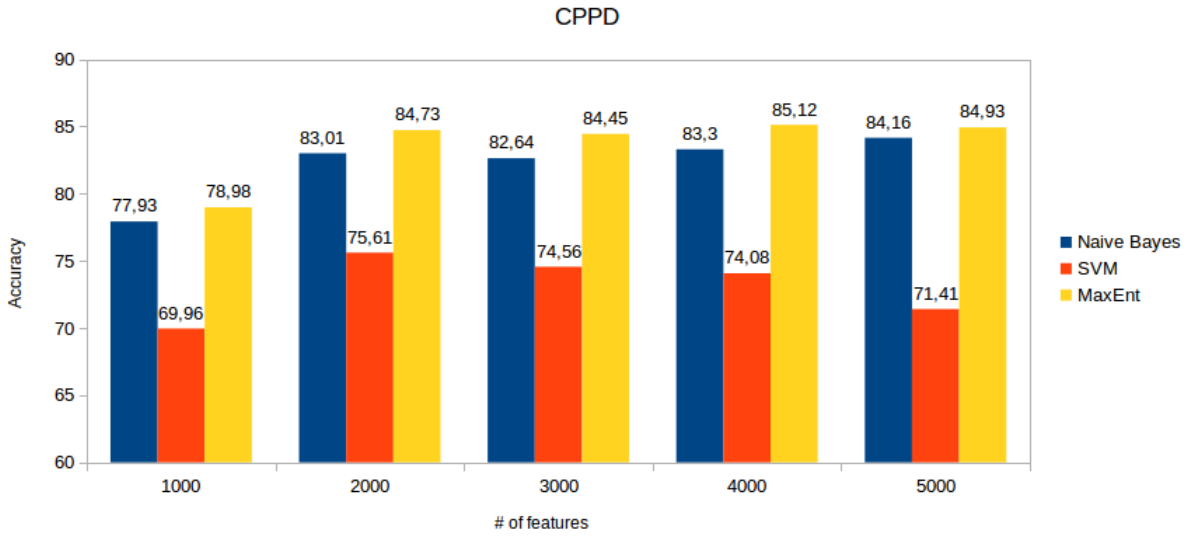

Figura 9 – Performance dos classificadores utilizando o método de seleção CPPD.

Tabela 12 – Desvios-padrão da acurácia medida em função do classificador e o número de atributos selecionados com a técnica CPPD

	Num. Atributos				
	1000	**2000**	**3000**	**4000**	**5000**
Naïve Bayes	4,9	3,6	3,8	4,5	3,1
MaxEnt	4,2	3,0	2,9	3,2	3,3
SVM	3,0	4,5	4,1	4,2	5,5

Conforme pode ser visto na Tabela 12, o modelo MaxEnt apresentou o menor desvio-padrão no intervalo entre 2000 e 4000 dimensões, significando que neste intervalo o classificador, além de possuir maior acurácia, é mais preciso na tarefa de classificação.

4.3 Comparação com outros trabalhos

Os resultados encontrados aqui nesse trabalho são comparáveis aos encontrados na literatura, conforme Tabela 13.

Tabela 13 – Tabela comparativa de desempenho em trabalhos semelhantes.

Autor (ano)	**Atributos**	**Técnica**	**# Classes**	**Acurácia(%)**
Este trabalho	*unibigrams*	CPPD, MaxEnt	3	85.12
Morgado (2012)	*unigrams*	*minimum cuts* léxico *Valence Shifters*	3	56.61
Martinazzo, Dosciatti e Paraiso (2011)	*unigram*	LSA	6	70.5
Alvim et al. (2010)	*bigrams*, *POS tags*	SVM	2	84

Os trabalhos mencionados como referências na Tabela 13 não se tratam da mesma aplicação de análise de sentimento em notícias políticas, porém, estão relacionados a classificação de notícias em geral. Isso se deve a falta de trabalhos relacionados a mesma aplicação, devido ao fato do campo de notícias políticas ser pouco explorado em aplicações de NLP no Brasil.

A metodologia apresentada neste trabalho tem como vantagem a limitação da quantidade de atributos sem comprometer a performance da classificação. O uso combinado de *unigrams* e *bigrams* selecionados dispensa o uso de técnicas mais sofisticadas e custosas computacionalmente, obtendo um desempenho comparável aos apresentados na literatura.

A aplicação da técnica CPPD somente é necessária na etapa de treinamento do algoritmo de classificação, onde será gerado um vocabulário para o modelo *bag-of-words*. De posse deste vocabulário, o CPPD só deve ser aplicado novamente em uma eventual necessidade de atualização do vocabulário. Diferentemente nos trabalhos mencionados na Tabela 13, técnicas como *minimum cuts* e a extração das POS *tags* de uma sentença exigem aplicação a cada novo texto a ser classificado. Ambas técnicas adicional uma fase

custosa de pré-processamento do texto que pode levar a explosão do número de dimensões sem um ganho equivalente de desempenho em termos de acurácia da classificação.

5 Conclusão

Visto a quantidade de aplicações de Análise de Sentimento voltadas para redes sociais, críticas de filmes e produtos, o campo de análise de notícias ainda se encontra em crescimento. Analisar notícias sobre as eleições podem gerar indicadores interessantes para previsão de resultados eleitorais.

O presente trabalho expôs um estudo comparativo do impacto de técnicas de seleção de atributos e classificadores baseados em aprendizagem de máquina para classificação de notícias políticas no período das eleições de 2014. Foi apresentada uma metodologia baseada na extração de atributos híbrida: *unigram* e *bigram* juntos sem o crescimento exponencial da quantidade de atributos.

Foi constatado que, para essa aplicação, a combinação do método CPPD de seleção de atributos e o classificador Regressão Logística obtiveram o melhor desempenho sobre as outras técnicas e classificadores escolhidos. Isso se deve ao fato do método CPPD além de selecionar os melhores atributos, ele também os organiza em ordem de relevância para uma determinada classe. Dessa forma, somente os atributos mais relevantes são escolhidos.

A baixa dimensionalidade da representação *bag-of-words* do texto é um diferencial. O desempenho do classificador não é gravemente afetado ao no intervalo entre 2000 e 5000 atributos selecionados. O pico de performance é alcançado ao selecionarmos 4000 atributos, entretanto a diferença entre as acurácias medidas nesse intervalo não chega a 1%. Portanto, é possível utilizar a representação de mais baixa dimensionalidade sem um comprometer a performance do classificador.

Este trabalho teve por objetivo explorar as possibilidades de classificação de notícias, especialmente no domínio político, para o português brasileiro. Visto que o campo de análise de notícias políticas é pouco explorado na literatura, este trabalho pretendeu fornecer resultados preliminares para servir de ponto de partida para aplicações mais sofisticadas como análise de viés jornalístico ou mesmo aplicações comerciais de análise de sentimento.

5.1 Perspectivas e Trabalhos Futuros

Um dos desafios encontrados no desenvolvimento deste trabalho foi a escassez de *corpus* disponíveis para este tipo de aplicação. Tendo em vista a continuação e evolução do modelo desenvolvido aqui, pretende-se coletar mais notícias no domínio da política brasileira não somente no âmbito das eleições. A coleta pode ser automatizada através do desenvolvimento *scripts* que realizem as requisições periodicamente do noticiário.

A coleta dos dados não constitui um desafio isolado, mas juntamente com ele encontramos o obstáculo da rotulação manual dos dados para treinamento dos modelos. A disponibilidade de pessoas para ler e avaliar cada notícia e rotular cada parágrafo torna o desafio maior. Um possível solução para essa problemática é o desenvolvimento de uma ferramenta colaborativa que permita as pessoas avaliarem a notícia dentro do próprio enquanto lêem o artigo em questão.

A correlação entre a opinião/sentimento observada nos noticiários e as redes sociais ao longo do tempo pode ser uma característica interessante afim de se investigar o quanto as notícias exercem influência sobre a opinião pública e, possivelmente, como antecipar a reação da população frente ao lançamento de uma notícia.

5.2 Trabalhos aceitos para publicação

CARVALHO, Caio Magno Aguiar; NAGANO, Hitoshi; BARROS, Allan Kardec. A Comparative Study for Sentiment Analysis on Election Brazilian News. In: Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology. 2017. p. 103-111.

Referências

AGARWAL, B.; MITTAL, N. Categorical probability proportion difference (cppd): a feature selection method for sentiment classification. In: *Proceedings of the 2nd workshop on sentiment analysis where AI meets psychology, COLING*. [S.l.: s.n.], 2012. p. 17–26. Citado na página 31.

ALVIM, L. et al. Sentiment of financial news: a natural language processing approach. In: *1st Workshop on Natural Language Processing Tools Applied to Discourse Analysis in Psychology, Buenos Aires*. [S.l.: s.n.], 2010. Citado 2 vezes nas páginas 16 e 42.

ARRUDA, G. D. de; ROMAN, N. T.; MONTEIRO, A. M. An annotated corpus for sentiment analysis in political news. In: *Proceedings of the 10th Brazilian Symposium in Information and Human Language Technology*. [S.l.: s.n.], 2015. p. 101–110. Citado 2 vezes nas páginas 17 e 33.

BERGER, A. L.; PIETRA, V. J. D.; PIETRA, S. A. D. A maximum entropy approach to natural language processing. *Computational linguistics*, MIT Press, v. 22, n. 1, p. 39–71, 1996. Citado na página 23.

BIRD, S.; KLEIN, E.; LOPER, E. *Natural language processing with Python: analyzing text with the natural language toolkit*. [S.l.]: "O'Reilly Media, Inc.", 2009. Citado na página 14.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. *Journal of machine Learning research*, v. 3, n. Jan, p. 993–1022, 2003. Citado na página 15.

CALAMBÁS, M. A. et al. Judicial precedents search supported by natural language processing and clustering. In: IEEE. *Computing Colombian Conference (10CCC), 2015 10th*. [S.l.], 2015. p. 372–377. Citado na página 14.

DOSCIATTI, M. M.; FERREIRA, L. P. C.; PARAISO, E. C. Identificando emoçoes em textos em português do brasil usando máquina de vetores de suporte em soluçao multiclasse. *ENIAC-Encontro Nacional de Inteligência Artificial e Computacional. Fortaleza, Brasil*, 2013. Citado 2 vezes nas páginas 16 e 17.

HADDI, E.; LIU, X.; SHI, Y. The role of text pre-processing in sentiment analysis. *Procedia Computer Science*, Elsevier, v. 17, p. 26–32, 2013. Citado na página 28.

HAYKIN, S. S. et al. *Neural networks and learning machines*. [S.l.]: Pearson Upper Saddle River, NJ, USA:, 2009. Citado na página 25.

IQBAL, F.; FUNG, B.; DEBBABI, M. Mining criminal networks from chat log. In: IEEE COMPUTER SOCIETY. *Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*. [S.l.], 2012. p. 332–337. Citado na página 14.

JOSE, R.; CHOORALIL, V. S. Prediction of election result by enhanced sentiment analysis on twitter data using word sense disambiguation. In: IEEE. *Control Communication & Computing India (ICCC), 2015 International Conference on*. [S.l.], 2015. p. 638–641. Citado 2 vezes nas páginas 14 e 15.

MARTINAZZO, B.; DOSCIATTI, M. M.; PARAISO, E. C. Identifying emotions in short texts for brazilian portuguese. In: *IV International Workshop on Web and Text Intelligence (WTI 2012)*. [S.l.: s.n.], 2011. Citado 2 vezes nas páginas 16 e 42.

MAYNARD, D.; FUNK, A. Automatic detection of political opinions in tweets. In: SPRINGER. *Extended Semantic Web Conference*. [S.l.], 2011. p. 88–99. Citado 2 vezes nas páginas 14 e 15.

MILLER, G. A. Wordnet: a lexical database for english. *Communications of the ACM*, ACM, v. 38, n. 11, p. 39–41, 1995. Citado na página 15.

MORAES, S. M.; MANSSOUR, I. H.; SILVEIRA, M. S. 7x1pt: um corpus extraído do twitter para análise de sentimentos em língua portuguesa. *BRACIS, STIL*, 2015. Citado na página 15.

MORGADO, I. C. Classification of sentiment polarity of portuguese on-line news. In: *Proceedings of the 7th Doctoral Symposium in Informatics Engineering*. [S.l.: s.n.], 2012. p. 139–150. Citado 2 vezes nas páginas 16 e 42.

PADMAJA, S.; FATIMA, S. S.; BANDU, S. Analysis of sentiment on newspaper quotations: A preliminary experiment. In: IEEE. *Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on*. [S.l.], 2013. p. 1–5. Citado na página 16.

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. Citado 2 vezes nas páginas 14 e 15.

PANG, B.; LEE, L.; VAITHYANATHAN, S. Thumbs up?: sentiment classification using machine learning techniques. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. [S.l.], 2002. p. 79–86. Citado 2 vezes nas páginas 15 e 21.

PARDO, T. A. et al. Computational linguistics in brazil: an overview. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas*. [S.l.], 2010. p. 1–7. Citado na página 15.

PASINATO, M. B.; MELLO, C. E.; aO, G. Z. Acompanhamento de campanha eleitoral pelo twitter. In: Bastos Filho, C. J. A.; POZO, A. R.; LOPES, H. S. (Ed.). *Anais do 12 Congresso Brasileiro de Inteligência Computacional*. Curitiba, PR: ABRICOM, 2015. p. 1–6. Citado na página 15.

RINGSQUANDL, M.; PETKOVIC, D. Analyzing political sentiment on twitter. In: *AAAI Spring Symposium: Analyzing Microtext*. [S.l.: s.n.], 2013. p. 40–47. Citado na página 15.

RITT, C. F.; WERLE, C. C. A influência que os meios de comunicação exercem na formação da opinião pública através do exercício exacerbado e sem limites da liberdade de expressão e à informação. *Anais do Salão de Ensino e de Extensão*, p. 199, 2013. Citado na página 14.

SCHOUTEN, K.; FRASINCAR, F. Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 28, n. 3, p. 813–830, 2016. Citado 2 vezes nas páginas 15 e 21.

SHARMA, A.; DEY, S. A comparative study of feature selection and machine learning techniques for sentiment analysis. In: ACM. *Proceedings of the 2012 ACM research in applied computation symposium*. [S.l.], 2012. p. 1–7. Citado na página 28.

SIMEON, M.; HILDERMAN, R. Categorical proportional difference: A feature selection method for text categorization. In: AUSTRALIAN COMPUTER SOCIETY, INC. *Proceedings of the 7th Australasian Data Mining Conference-Volume 87*. [S.l.], 2008. p. 201–208. Citado na página 29.

TAUSCZIK, Y. R.; PENNEBAKER, J. W. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, Sage Publications Sage CA: Los Angeles, CA, v. 29, n. 1, p. 24–54, 2010. Citado na página 15.

YANG, K.-S. et al. Name entity extraction based on pos tagging for criminal information analysis and relation visualization. In: IEEE. *Information Science and Service Science and Data Mining (ISSDM), 2012 6th International Conference on New Trends in*. [S.l.], 2012. p. 785–789. Citado 2 vezes nas páginas 14 e 28.