UNIVERSIDADE ESTADUAL DE CAMPINAS
Instituto de Computação

Rafael Medeiros Jacomel de Oliveira Silva

Análise de Sentimento em *tweets*

Campinas 2018

Rafael Medeiros Jacomel de Oliveira Silva

Análise de Sentimento em *tweets*

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Ciência de Computação.

Orientador: Professor Dr. Jacques Wainer

Este exemplar corresponde à versão final da dissertação defendida pelo aluno Rafael Medeiros Jacomel de Oliveira Silva, e orientada pelo prof. Dr. Jacques Wainer

Campinas 2018

Agência(s) de fomento e nº(s) de processo(s): Não se aplica.

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

Si38a

Silva, Rafael Medeiros Jacomel de Oliveira, 1981-
Análise de sentimento em tweets / Rafael Medeiros Jacomel de Oliveira Silva. – Campinas, SP : [s.n.], 2018.

Orientador: Jacques Wainer.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Redes sociais on-line. 2. Twitter (Rede social on-line). 3. Emoções - Classificação. 4. Linguística - Processamento de dados. 5. Processamento da linguagem natural (Computação). I. Wainer, Jacques, 1958-. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Sentiment analysis in tweets
Palavras-chave em inglês:
Online social networks
Twitter (Social network online)
Emotions - Classification
Linguistics - Data processing
Natural language processing (Computer science)
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Jacques Wainer [Orientador]
Sandra Eliza Fontes de Avila
Nádia Félix Felipe da SIlva
Data de defesa: 05-04-2018
Programa de Pós-Graduação: Ciência da Computação

COMISSÃO JULGADORA - DISSERTAÇÃO DE MESTRADO

Candidato: Rafael Medeiros Jacomel de Oliveira Silva

RA: 140270

Data da defesa: 5 de Abril de 2018.

Título da dissertação: Análise de Sentimento em *tweets*

Comissão Julgadora:

Prof. Dr. Jacques Wainer (Presidente, IC/UNICAMP)

Prof[a] Dra. Sandra Eliza Fontes de Avila (IC/UNICAMP)

Prof[a] Dra. Nádia Félix Felipe da Silva (INF/UFG)

A ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no processo de vida acadêmica do aluno.

Agradecimentos

Agradeço meus familiares, que sempre me suportaram, encorajaram e me ajudaram a continuar estudando e melhorando como pessoa e profissional. Agradeço a IBM Brasil por me incentivar a continuar os estudos. Agradeço a Unicamp, em especial ao professor Jacques Wainer pela atenção e pelo suporte e que me aceitou como aluno mesmo eu trabalhando e estudando, condição diferenciada dos demais alunos.

Agradeço aos amigos Daniel Cárnio Junqueira e João Alexandre Ferreira da Rocha Pereira, o primeiro por contribuir com a revisão desse trabalho e o segundo por me incentivar fortemente nessa jornada e que, apesar de não ver mais, estará sempre na minha memória.

Obrigado a todos por permitirem que isso fosse possível.

Resumo

Análise do sentimento é um campo de estudo de recente popularização devido ao crescimento da *Internet* e ao conteúdo gerado por seus usuários. Mais recentemente, as redes sociais surgiram, nessas redes as pessoas publicam suas opiniões em linguagem coloquial e compacta. Isto é o que acontece, por exemplo, no *Twitter*, uma ferramenta de comunicação que pode ser facilmente utilizada como fonte de informação para várias ferramentas automatizadas de inferência de sentimento. Esforços de pesquisa foram direcionados para lidar com o problema da análise do sentimento nas redes sociais do ponto de vista de um problema de classificação, onde não há consenso sobre qual é o melhor classificador, qual a melhor forma de pré-processamento entre outros. O objetivo desta dissertação é investigar a influência de algumas técnicas de pré-processamento, da técnica TF-IDF, do volume do conjunto de treinamento e de técnicas *ensembles* na acurácia de alguns classificadores supervisionados.

Abstract

Sentiment analysis is a field of study that shows recent popularization due to the growth of Internet and the content that is generated by its users. More recently, social networks have emerged, where people post their opinions in colloquial and compact language. This is what happens in Twitter, a communication tool that can easily be used as a source of information for various automatic tools of sentiment inference. Research efforts have been directed to deal with the problem of sentiment analysis in social networks from the point of view of a classification problem, where there is no consensus about what the best classifier is, and what is the best configuration provided by the feature engineering process. The objective of this dissertation is to investigate the influence of some pre-processing techniques, the TF-IDF technique, the volume of the training set and ensembles techniques in the accuracy of some supervised techniques.

Lista de Tabelas

Sumário

Capítulo 1
Introdução

Análise de sentimento, também chamada de mineração de opinião, é o campo de estudo que analisa opiniões, sentimentos, avaliações, atitudes e emoções de pessoas sobre produtos, serviços, organizações, indivíduos, eventos, tópicos e seus atributos. O problema possui um amplo espaço para evolução. Existem, também, muitos nomes e tarefas ligeiramente diferentes como, por exemplo, mineração de sentimentos, mineração de opinião, extração de opinião, análise de subjetividade, análise de afeto, análise de emoção, mineração de avaliação, etc. Entretanto, todos esses termos são muitas vezes utilizados como equivalentes à tópicos de análise de sentimento ou mineração de opinião [1].

O termo análise de sentimento, apareceu pela primeira vez em [2], e o termo mineração opinião em [3]. No entanto, a investigação sobre os sentimentos e opiniões apareceu mais cedo em [4] [5] [6] [7] [8] [9]. Neste trabalho, utilizaremos os termos análise de sentimento e mineração de opinião como sinônimos e os distinguiremos quando necessário.

Existem diversos motivos pelos quais classificar *tweets* de acordo com um sentimento específico se tornou importante. Tal classificação possui muitas aplicações na ciência política, nas ciências sociais, na pesquisa de mercado e em muitos outros [10]. Em segundo lugar, existem desafios de pesquisa estabelecidos quanto ao tipo de fonte textual a ser tratada, uma vez que resultados de análises de sentimentos mostram que diferentes tipos de textos requerem métodos especializados de análise como, por exemplo, sentimentos não são expressos da mesma maneira em textos jornalísticos, *blogs*, comentários, fóruns, mensagens em redes sociais ou outros tipos de conteúdos gerados por usuários [11].

Apesar da pesquisa em análise de sentimento começar a partir do início dos anos 2000 [2], alguns trabalhos anteriores exploraram a interpretação de metáforas, adjetivos, sentimento, subjetividade, pontos de vista e afeto [11] [12] [13] [14] [15] [16].

1.1 Aplicações de Análise de Sentimento

Opiniões são centrais para quase todas as atividades humanas, porque elas são os principais influenciadores do nosso comportamento. Sempre que precisamos tomar alguma decisão, nós queremos saber a opinião dos outros. No mundo real, as empresas e organizações estão sempre preocupadas em saber a opinião pública sobre os seus produtos e serviços. Os consumidores individuais também querem saber as opiniões dos usuários existentes de um produto antes de comprá-lo e eleitores buscam opiniões de outros sobre candidatos políticos antes de tomar uma decisão de voto em uma eleição política [1].

Algumas aplicações práticas de análise de sentimento foram propostas, por exemplo em [104], onde foi proposto um modelo de análise de sentimento para prever o desempenho de vendas; em [105] comentários foram utilizados para classificar produtos e comerciantes; em [22], as relações entre a linha de apostas e opiniões públicas em *blogs* e *Twitter* foram estudadas; em [106], em que opiniões expressas no *Twitter* foram cruzadas com pesquisas de opinião pública.

Em [107], as opiniões expressas no *Twitter* foram aplicadas para prever resultados eleitorais. Em [108], os autores estudaram pontos de vista políticos. Em [109], um método para a previsão de comentários de *blogs* políticos foi desenvolvido. Em [110] dados de crítica coletados no *Twitter* foram utilizados para prever a receita de bilheteria para filmes.

Em [17], o fluxo de sentimento em redes sociais foi investigado. Em [18], sentimentos de *e-mails* foram utilizados para descobrir como os sexos diferem ao expressar suas emoções. Em [19], as emoções expressas em romances e contos de fadas foram rastreadas.

Em [20], opiniões expressas no *Twitter* foram utilizadas para prever o mercado de ações. Em [21], investidores experientes em *microblogs* foram identificados e análise de sentimento de suas ações foram realizadas. Em [22], um *blog* e os sentimento expressos em notícias foram utilizados para estudar estratégias de negociação.

Em [23], as influências sociais em resenhas de livros foram estudadas. Em [24], a análise de sentimento foi utilizada para caracterizar as relações sociais. Um sistema de análise de sentimento abrangente e alguns estudos de casos também foram relatados em [25].

1.2 Pesquisa em Análise de Sentimento

Como discutido anteriormente, as aplicações da vida real são apenas parte da razão pela qual análise de sentimento é um problema de pesquisa popular. O tema é altamente desafiador como um tema de pesquisa em PLN (Processamento de Linguagem Natural) e abrange muitos problemas novos. Além disso, houve pouca pesquisa antes do ano 2000 nessa área [1].

Uma das razões é que antes disso havia pouco texto de opinião disponíveis em forma digital. Desde o ano 2000, o campo tem crescido rapidamente para se tornar uma das áreas de pesquisa mais ativas em PLN. É também amplamente pesquisada em mineração de dados, mineração *Web* e em recuperação de informação. O tema se espalhou desde a área de ciência da computação até as ciências da gestão [26] [27] [28] [29] [30] [31].

1.2.1 Níveis de Análise

Daremos uma breve introdução aos principais problemas de pesquisa com base no nível de granularidade da pesquisa existente. Em geral, análise de sentimento tem sido investigada principalmente em três níveis de granularidade:

Nível do documento: A tarefa neste nível é classificar se uma opinião inteira expressa em um documento se trata de um sentimento positivo ou negativo [6]. Por exemplo, dada uma revisão do produto, o sistema determina se a revisão expressa uma opinião positiva ou negativa geral sobre o produto.

Nível de sentença: A tarefa nesse nível determina se cada frase expressa um sentimento positivo, negativo ou neutro. Neutro geralmente significa ausência de sentimento. O nível de análise está intimamente relacionado com a subjetividade [16], que distingue frases objetivas (frases que expressam informação factual) de frases subjetivas (que expressam visões subjetivas e opiniões). No entanto, devemos observar que a subjetividade não é equivalente ao sentimento já que muitas frases objetivas podem implicar opiniões [16].

Nível de entidade e aspecto: Tanto a nível de documento quanto no nível de frase podem não descobrir o que exatamente as pessoas gostaram ou não gostaram [1]. Nível de aspecto realiza uma análise mais refinada. Nível de aspecto foi anteriormente chamada de nível de recurso [30]. Em vez de olhar para construções de linguagem (documentos, nível de aspecto

olha diretamente para a própria opinião. Baseia-se na ideia de que uma opinião é constituída por um sentimento (positivo ou negativo) e um destino (de opinião). Uma opinião sem o seu alvo ser identificado é de uso limitado [30].

Perceber a importância de alvos de uma opinião também nos ajuda a entender melhor o problema de análise de sentimento. Por exemplo, embora a frase "embora o serviço não seja tão bom, eu ainda amo este restaurante" tenha claramente um tom positivo, não podemos dizer que esta frase é inteiramente positiva. Na verdade, a sentença é positiva sobre o restaurante (enfatizado), mas negativa sobre o seu serviço (não é enfatizado).

Por exemplo, a frase "a qualidade da chamada do *iPhone* é boa, mas a vida da bateria é curta", avalia dois aspectos, a qualidade da chamada e vida útil da bateria do *iPhone* (entidade). O sentimento a respeito da qualidade da chamada de *iPhone* é positivo, mas o sentimento sobre a vida útil da bateria é negativo. A qualidade da chamada e vida da bateria do *iPhone* são os alvos de opinião.

Com base nesta análise de nível, um resumo estruturado de opiniões sobre entidades e seus aspectos pode ser produzido. Ambas as classificações, as de nível de documento e as de nível de sentença já são altamente desafiadoras.

1.3 Contribuição e Organização da Dissertação

Nesse trabalho avaliamos a influência de algumas técnicas de pré-processamento e da técnica *Term Frequency–Inverse Document Frequency* (TF-IDF) na acurácia de alguns classificadores supervisionados no contexto de análise de sentimento.

Analisamos, também, como o volume do conjunto de treinamento e técnicas *ensembles* influenciam na acurácia dos mesmos classificadores supervisionados. Os classificadores supervisionados utilizados foram o *Support Vectors Machine* (SVM), *Random Forest*, *Logistic Regression* e *Naïve Bayes.*

A análise de sentimento em todas as situações foi feita utilizando dois níveis de sentimento (positivo e negativo) no nível de sentença. Vale ressaltar que, para a tarefa da análise de sentimentos em *tweets,* não há diferença fundamental entre o nível de documento e o nível de sentença [32].

O restante dessa dissertação está organizado da seguinte forma: No Capítulo 2 apresentamos alguns trabalhos relacionados e métricas de avaliação. No Capítulo 3 falaremos sobre a metodologia proposta nesse trabalho. No Capítulo 4 descreveremos os experimentos realizados e discutiremos os resultados. No Capítulo 5 concluiremos o trabalho e falaremos sobre possíveis trabalhos futuros.

Capítulo 2
Trabalhos Relacionados e Métricas de Avaliação

Nesse Capítulo descreveremos algumas abordagens utilizadas para análise de sentimentos em *Twitter* e algumas métricas de avaliação. Em [32] o autor descreve as principais abordagens da literatura para classificar sentimentos em *tweets*. O autor menciona 4 métodos principais, encontrados na literatura, para classificar *tweets*:

- Aprendizado de máquina.
- Léxicos.
- Métodos Híbridos de aprendizado de máquina e léxicos.
- Métodos baseados em grafos.

Descreveremos, também, algumas das mais populares métricas de avaliação de desempenho: Acurácia, Precisão, *Recall*, *F-measure*.

2.1 Métodos Baseados em Aprendizado de Máquina

A maioria dos métodos propostos que lidam com análise de sentimentos em *Twitter* empregam algum classificador de aprendizado de máquina que é treinado com alguns atributos de *tweets*. Nessa seção iremos revisar alguns trabalhos que utilizaram aprendizado de máquina na tarefa de análise de sentimentos em *Twitter*.

Alguns dos classificadores mais aplicados são *Naïve Bayes* (NB), Entropia Máxima (MaxEnt), *Support Vectors Machine* (SVM), *Multinomial Naïve Bayes* (MNB), *Logistic Regression* (LR), *Random Forest* (RF) e *Conditional Random Field* (CRF) [32].

2.1.1 Aprendizado Supervisionado

Um dos classificadores mais utilizados para endereçar o problema de análise de sentimentos em *Twitter* é o classificador SVM, em [33] foi utilizado o classificador SVM no conjunto de dados fornecido pelo *workshop* SemEval-2013 [34]. Os autores representaram cada *tweet* como um vetor de atributos que incluía, palavras em maiúsculas, *hashtags*, léxicos, pontuação, *emoticons*, alongamento enfático e negação. Eles observaram que o classificador SVM treinado utilizando esses recursos tem desempenho melhor do que os treinados apenas com *unigrams*. O método obteve um *F-measure* de 69,02% para no nível de frase. Os autores concluíram que as características mais úteis na tarefa de análise de sentimentos são as características de léxicos e os *n-grams*.

Em [37] foi proposto utilizar muitos recursos com o objetivo de conseguir um bom desempenho na tarefa de análise de sentimentos. Os atributos examinados incluíram conceitos de *DBPedia* [37], grupos de verbo e adjetivos da *WordNet* e *senti-features* de *Senti-WordNet*. Em [37] foi empregado um dicionário de emoções e abreviaturas para melhorar o desempenho nessa tarefa. O método conseguiu melhorar a precisão da medida *F-measure* em 2% em relação ao SVM treinado com *unigrams* e em 4% em relação ao classificador NB.

Em [35] os autores investigaram como as mudanças nos preços das ações de uma empresa, (aumentos e quedas), estão correlacionadas com as opiniões públicas expressas em *tweets* sobre essa empresa. O presente trabalho empregou duas representações textuais diferentes (*Word2vec* e *n-gram)* para analisar os sentimentos do público em *tweets*. Foram utilizados, também, os classificadores *Random Forest* e *Logistic Regression*.

Foi concluído que notícias positivas e *tweets* nas mídias sociais sobre uma empresa definitivamente encorajariam as pessoas a investir nas ações daquela empresa e, como resultado, o preço das ações daquela empresa aumentaria. No final do artigo, mostra-se que existe uma forte correlação entre os aumentos e quedas nos preços das ações com a polaridade nos sentimentos do público expressos em *tweets*.

Em [36] os autores descrevem como construir modelos *Word2Vec* utilizando um extenso conjunto de dados na língua árabe obtido de *tweets* relacionados a serviços de saúde. O modelo proposto utilizou a arquitetura CBOW (*Continuous Bag of Words*) com 200 dimensões para construir seu modelo de *Word2Vec*. A arquitetura CBOW consiste em prever as palavras centrais ou palavras alvo das palavras adjacentes dentro de um comprimento de

janela.

Foram utilizados os algoritmos supervisionados: MNB (*Multinomial Naïve Bayes*), NSVC (*Nu-Support Vector Classification*), LSVC (*Linear Support Vector Classification*), LR (*Logistic Regression),* SGD *(Stochastic Gradient Descent*) e RDG (*Ridge Classifier*).

O melhor resultado da abordagem mencionada acima melhorou o resultado em 7 pontos percentuais (conseguiu acurácia de 92% contra 85% de abordagens anteriores) na base de dados própria utilizada.

A tabela 2.1 resume os artigos que mencionamos e que empregaram algum método supervisionado para abordar a tarefa de análise de sentimentos. A primeira coluna mostra informações para a referência, a segunda coluna mostra o ano de publicação do trabalho, a terceira coluna mostra os algoritmos utilizados em cada estudo, a quarta coluna mostra os atributos e a ultima coluna mostra o conjunto de dados.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Mohammad et al. [33]	2013	SVM	*n-grams* de palavras, caracteres, POS, caps, léxicon, pontuação, negação	SemEval- 2013
Hamdan et al. [37]	2013	SVM, NB	*Unigrams*, DBPedia, WordNet, SentiWordNet	SemEval- 2013
Pagolu, V., Challa, K. and Panda, G. [35]	2016	LR, RF	*Word2vec, n-gram*	Próprio
Alayba, A., Palade, V., England, M. and Iqbal, R. [36]	2018	MNB, NSVC, LSVC, LR, SGD, RGD.	*Word2vec*	Próprio

Tabela 2.1: Sumário de artigos que empregaram métodos supervisionados na tarefa de análise de sentimento.

2.1.2 Classificadores *Ensembles*

Recentemente, o conceito de combinar classificadores foi proposto como uma nova direção para melhorar o desempenho de classificadores individuais [32]. Esta abordagem, conhecida como *ensembles* de classificadores, também tem sido aplicada na tarefa de análise de sentimento [32]. Abaixo descreveremos alguns autores que utilizaram essa abordagem em seus trabalhos.

Classificadores *ensembles* foram aplicados para endereçar a tarefa de análise de sentimento no nível de frase em [38] [39]. Em [38] os autores apresentaram uma abordagem que classifica automaticamente o sentimento de *tweets* utilizando conjuntos de classificadores e léxicos. Os *tweets* são classificados como positivos ou negativos em relação a um termo de consulta.

Foram utilizadas as técnicas de *Bag-of-Words* (BoW) e *Feature Hashing* para representação de *tweets*. Na fase de pré-processamento *retweets*, *stop words*, *links*, URLs, menções, pontuação e acentuação foram removidos para que o conjunto de dados pudesse ser padronizado. Foi utilizado *stemming* para que a matriz ficasse menos esparsa. O *BoW* foi construído com frequência binária, e um termo é considerado "frequente" se ocorrer em mais de um *tweet*.

Foram utilizados léxicos utilizando uma lista de 4.783 palavras negativas e 2.006 palavras positivas [38]. Essa lista foi compilada ao longo de muitos anos e cada uma das suas palavras indica uma opinião. As palavras com conotação positiva são utilizadas para expressar os estados desejados, enquanto as palavras de conotação negativa são utilizadas para expressar estados indesejados.

Emoticons disponíveis em *tweets* foram utilizados para enriquecer o conjunto de atributos. O número de *emoticons* positivos e negativos foi utilizado para complementar a informação fornecida pelo *BoW* e o *feature hashing*. Além disso, foram calculados o número de léxicos positivos e negativos em cada mensagem.

Foram utilizados quatro conjuntos de dados cujos detalhes são descritos em [38]: *Sanders - Twitter Sentiment Corpus, Stanford - Twitter Sentiment Corpus*, *Obama-McCain Debate* (OMD) e o *Health Care Reform* (HCR).

Nesse trabalho foi demonstrado que conjuntos de classificadores formados por componentes diversificados podem fornecer bons resultados para esse domínio particular. Foram comparadas diferentes estratégias para a representação de *tweets* (*bag-of-words* e *feature hashing*) que mostraram suas vantagens e desvantagens.

O *feature hashing* mostrou ser uma boa escolha no cenário da análise do sentimento em *tweets*, onde o esforço computacional é de suma importância. Foi demonstrado, também, que quando o foco está na precisão, a melhor escolha é a técnica *bag-of-words*.

Em [39] os autores apresentaram dois classificadores *ensembles* elaborados a partir de quatro classificadores: *MaxEnt* (Máxima Entropia) e *Multinomial Naïve Bayes* do *kit* de ferramentas de aprendizado de máquina *Mallet* [11], *SentiStrength* [11] e *Pattern2* [117].

Os autores avaliaram suas técnicas *ensembles* e os quatro classificadores individualmente em doze bases de dados de *Twitter*. Os autores demonstraram que os *ensembles* de classificadores propostos conseguiram o melhor desempenho nos experimentos realizados.

Em [40] o autor explorou a eficácia de diferentes versões do classificador *Perceptron*. Os seus resultados conseguiram melhorar a performance do classificador SVM quando utilizado o método de votação para classificar um *tweet*.

A tabela 2.2 resume os artigos que mencionamos e que empregaram algum método *ensemble* para abordar a tarefa de análise de sentimentos. A primeira coluna mostra informações para a referência, a segunda coluna mostra o ano de publicação do trabalho, a terceira coluna mostra os algoritmos utilizados em cada estudo, a quarta coluna mostra os atributos e a ultima coluna mostra o conjunto de dados.

Estudo	**Ano**	**Algoritmos**	**Atributos**	**Conjunto de dados**
da Silva, Hruschka and Hruschka [38]	2014	SVM, LR, NB, RF, *Ensembles*	léxicos, polaridade	*Sanders*, STS, OMD, HCR
Yang and Yan [39]	2017	*MaxEnt, Multinomial Naïve Bayes*	Atributos considerados pelo *MaxEnt*	STS-Gold, *Sanders*, Semval-2013, próprio
Aston et al. [40]	2014	*Perceptron*, *perception* por votação, *ensembles*	*n-grams* de caracteres	*Sanders*

Tabela 2.2: Sumário de artigos que empregaram métodos *ensembles* na tarefa de análise de sentimento.

2.1.3 *Deep Learning*

Deep learning é um dos campos que vem crescendo mais rapidamente na área de aprendizado de máquina, essa técnica tem sido frequentemente aplicada para resolver problemas relacionados a reconhecimento de imagem e processamento de linguagem natural. *Deep learning* utiliza redes neurais para aprender diversos níveis de abstração. Em tarefas relacionadas a texto, as abordagens de *deep learning* incluem, tipicamente, duas etapas.

Primeiro, a rede aprende representações numéricas das palavras que são utilizadas para representar documentos. No contexto de análise de sentimento, *deep learning* é utilizado para aprender representações numéricas de palavras em grandes quantidades de dados de texto [41]. Recentemente, em [42] os autores utilizaram *deep learning* para aprender representações semânticas de usuários e produtos, enquanto em [43] os autores utilizaram a técnica para previsão em revisão de produtos.

Deep Learning também foi explorado na tarefa de análise de sentimento em [44] onde os autores criaram uma solução em duas fases. Na primeira fase, foi classificado o que o autor chamou de subjetividade, ou seja, se um determinado *tweet* é neutro ou subjetivo em relação ao tópico dado.

Na segunda fase, foi classificado o sentimento dos *tweets* subjetivos (ignorando os *tweets* neutros), ou seja, se um determinado *tweet* subjetivo tem uma conotação negativa ou positiva em relação ao tópico. Uma rede neural profunda baseada em LSTM (*Long Short-Term memory*) foi proposta para cada fase.

O *dataset* utilizado foi SemEval 2016 [44], e foi obtida um *F-measure* de 68.84% e uma acurácia de 60.2% no melhor resultado dos experimentos, superando as soluções baseadas em *deep leaning* existentes para base de dados utilizada até o momento.

Em [45] os autores propuseram uma rede neural recursiva adaptativa (AdaRNN) para fazer análise de sentimento no nível de entidade. Esse método utilizou uma árvore de dependência para encontrar palavras sintaticamente relacionadas a uma dada entidade alvo e relacionar o sentimento dessas palavras para as entidades. O AdaRNN foi avaliado em um conjunto de dados anotado manualmente consistindo em 6248 *tweets* para treinamentos e 692 *tweets* para teste e conseguiu obter *F-measure* de 65,9%.

Em [46] os autores mencionam o problema da análise de sentimento em diferentes línguas. Segundo o autor, tal problema pode ser custoso computacionalmente falando e pode proporcionar resultados de baixa qualidade. Considerando esse cenário o autor propõe um tradutor eficiente com arquitetura baseada em rede neural para análise de sentimento em diferentes línguas.

No estudo foi utilizado um conjunto de dados do *Twitter* contendo dados marcados como positivo, negativo ou neutro de 13 línguas europeias diferentes [46]. Os autores comparam quatro arquiteturas baseadas em *deep learning* diferentes além de uma arquitetura baseada em SVM.

Três das arquiteturas baseadas em *deep learning* mencionadas são baseadas em CNN (*Convolutional Neural Network*) e o autor se refere a elas como Conv-Emb*[-Freeze]*, Conv-Char, Conv-Char-R. Uma das arquiteturas mencionadas se baseia em LSTM (*Long short-term memory*) e o autor se refere a mesma como LSTM-Emb. Tais arquiteturas utilizam *embeddings* no nível de palavra e no nível de caractere e são descritas em detalhes pelos autores em [46].

Arquiteturas baseadas em SVM também foram utilizadas. Foram criadas arquiteturas utilizando apenas *unigrams* (SVM-U), apenas *bigrams* (SVM-b), apenas *trigrams* (SVM-t), combinações de *unigrams* e *bigrams* (SVM-UB) e combinações de *unigrams*, *bigrams* e *trigrams* (SVM-UBT). Para avaliação dos modelos foram utilizadas as métricas de *F-measure* e acurácia.

Os resultados indicam que a arquitetura Conv-Char-R alcança resultados competitivos quando comparados com os modelos baseados em *deep learning* que alcançaram o estado da arte no problema de análise de sentimentos, com a principal vantagem, de consumir cerca de quatro vezes menos memória.

O conjunto de dados criado em [45] também foi utilizado em [47], nesse trabalho os autores propuseram usar um rico conjunto de aspectos. Na abordagem utilizada o *tweet* é dividido, em relação a uma entidade especifica, a esquerda e a direita. As representações numéricas das palavras foram utilizadas para modelar as interações dos dois contextos que foram utilizadas para detectar o sentimento relativo a entidade. Os autores exploraram uma série de funções de *pooling* para extrair automaticamente os aspectos. Sua abordagem superou o AdaRNN, obtendo um *F-measure* de 69,9%.

A tabela 2.3 resume os artigos que empregaram *deep learning* para resolver a tarefa de análise de sentimento.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Dey, K., Shrivastava, R., Kaushik, S. [44]	2018	LSTM	*Embeddings* de palavras	SemEval- 2016
Becker, W., Wehrmann, J., Cagnini and H., Barros, R. [46]	2017	SVM, LSTM, CNN	*Embeddings* de palavras, *embeddings* de caracteres, *unigram*, *bigram* e *trigram*	Dados de *twitter* de 13 línguas européias diferentes [46]
Dong et al. [45]	2014	AdaRNN	Arvore de dependência, *unigrams*, *bigrams*	próprio
Dui-Tin e Zhang [47]	2015	Target-ind, Target-dep	Léxicos, representação numérica de palavras, funções de *pooling*	O utilizado em [45]

Tabela 2.3: Sumário de artigos que empregaram métodos *deep learning* na tarefa de análise de sentimento.

2.2 Métodos Baseados em Léxicos

Métodos baseados em léxicos utilizam listas de palavras marcadas por polaridade ou pontuação de polaridade para determinar a pontuação geral de opinião de um determinado texto. A principal vantagem desses métodos é que eles não precisam de dados de treinamento. As abordagens baseadas em léxicos têm sido extensivamente aplicadas em textos convencionais, tais como *blogs*, fóruns e revisão de produtos [8] [115] [116].

No entanto, eles são menos explorados em análise de sentimento em comparação com métodos de aprendizado de máquina. A principal razão esta relacionada com a singularidade do texto do *Twitter* que não só contém um grande número de peculiaridades

textuais e expressões coloquiais, mas também tem uma natureza dinâmica com novas expressões e *hashtags* emergentes de tempos em tempos.

Um dos mais conhecidos algoritmos baseados em léxicos desenvolvidos para mídias sociais é o *SentiStrength* [48]. Utilizando o algoritmo *SentiStrength* consegue-se, efetivamente, identificar a polaridade do sentimento em um texto informal incluindo *tweets* utilizando léxicos codificados por humanos que contém palavras e frases que são frequentemente encontradas em mídias sociais.

Além de léxicos que contém cerca de 700 palavras, o *SentiStrength* utiliza uma lista de *emoticons*, negações e outras palavras para atribuir o sentimento a um texto. Inicialmente, o algoritmo foi testado nos comentários da rede social *MySpace*. O algoritmo foi estendido por [49] através da introdução de listas idiomáticas, novos léxicos e pela utilização de alongamento enfático. O *SentiStrength* foi comparado com muitas abordagens de aprendizado de máquina e testado em seis conjuntos de dados diferentes, incluindo um conjunto de dados com *posts* de *tweets*.

O conjunto de dados fornecidos pelo evento SemEval-2013 também foi utilizado por [50] para avaliar um sistema baseado em regras. Seu sistema era baseado em regras manuscritas, cada uma das quais tinha a forma padrão. Este sistema funcionou muito bem na tarefa de análise de sentimento e foi um dos sistemas de alto desempenho no evento SemEval-2013.

Um método interessante foi apresentado em [57], nesse trabalho os autores propuseram um método de análise do sentimento não-supervisionado baseado em sinais de emoção. Os sinais de emoção foram divididos em duas categorias: correlação de emoção e indicação de emoção. A abordagem de sinais emocionais para análise de sentimentos não supervisionados foi construída sobre o modelo de tri fatoração de matriz não-negativa ortogonal. Foram utilizados dois conjuntos de dados diferentes para a avaliação, os conjuntos de dados de STS [51] e o de OMD [52]. Experiências indicaram a eficácia do método proposto, bem como os papéis de diferentes sinais de emoção na análise do sentimento.

Em [53] os autores apresentaram o *SentiCircles*, uma abordagem baseada em léxicos para abordar a tarefa de análise do sentimento. O *SentiCircles* atualiza as pontuações atribuídas a polaridade das palavras em léxicos considerando os padrões de palavras que ocorrem juntamente em diferentes contextos. A abordagem *SentiCircles* foi avaliada em três

conjuntos de dados diferentes: OMD [52], HCR [54] e STS-Gold [55]. Experiências provaram a eficácia do método que superou os métodos baseados em *SentiWordNet*.

Uma metodologia para criar léxicos foi proposta em [56]. Os autores chamaram tal metodologia de *Senti-N-Gram*. A abordagem proposta é baseada em regras que são descritas, em detalhes, no trabalho mencionado. Basicamente, as regras geram pontuações de *unigrams*, *bigram* e *trigrams* em cinco níveis de sentimento (muito positivo, levemente positivo, neutro, levemente negativo, muito negativo). Tal abordagem utilizou um conjunto de dados que contém análises de produtos para criar essa escala de *n-gram*. O autor compara as pontuações desse método com as opiniões fornecidos por seres humanos utilizando *t-test* e as abordagens foram consideradas estatisticamente equivalentes.

O autor também propõe uma abordagem de classificação de sentimentos em positivo e negativo utilizando uma abordagem baseada em contagens de sentenças positivas e negativas (*unigrams*, *bigrams* e *trigrams*) de um documento. Essa abordagem, utilizando a metodologia *Senti-N-Gram,* foi comparada com duas abordagens existentes: uma abordagem baseada em *unigrams* VADER e outra baseada em *n-gramas* SOCAL [56]. Tal comparação foi feita utilizando dois conjuntos de dados de *benchmark* e a metodologia proposta pelo autor mostrou melhor desempenho em termos de acurácia, precisão, *recall* e *F-measure*.

A Tabela 2.4 resume os artigos que empregaram um método baseado em léxicos para tratar a tarefa de análise de sentimento.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Thelwall et al. [49]	2012	SentiStrength	Polaridade, *emoticons*, negação, alongamento enfático, *wordnet*, *sentiwordnet*.	SS-Tweet
Reckman et al. [50]	2013	Baseado em regras		SemEval- 2013
Hu et al. [57]	2013	ESSA	*emoticons*, léxicos, co-ocorrência de palavras.	STS, OMD
Saif et al. [53]	2016	SentiCircles	*SentiWordNet*, MPQA, léxicos.	OMD, HCR, STS-Gold
Dey, A., Jenamani, J. and Thakkar, J [56]	2018	Baseado em regras (*Senti-N-Gram*)		próprio

Tabela 2.4: Sumário de artigos que empregaram métodos de léxicos na tarefa de análise de sentimento.

2.3 Métodos Híbridos

Alguns pesquisadores têm combinado abordagens baseadas em léxicos com métodos de aprendizado de máquina. Em [58], os autores propuseram um método híbrido para endereçar a tarefa de análise de sentimento. Substituição de expressões negativas foram utilizadas (expressões como “won’t” foram substituídas por “will not”), foi feita remoção de URLs, reversão de palavras que contêm letras repetitivas a sua forma original em inglês foi utilizada **(**palavras como “cooooool” seriam substituída por “coool”), remoção de números e *stop words*.

Os autores utilizaram *unigrams* e *bigrams* e um modelo de pontuação de polaridade anterior baseado em AFFIN4 e *SentiWordNet*, tal pontuação é feita utilizando a soma de palavras do *tweet* com pontuação positiva menos a soma de palavras com pontuação negativa.

Os classificadores utilizados foram o SVM, *Naïve Bayes*, *Logistic Regression* e *Random Forest*. As bases utilizadas foram as bases de *Stanford Twitter Sentiment Test* (STS-Test), *SemEval2014*, Stanford *Twitter Sentiment Gold* (STS-Gold), *Sentiment Strength Twitter Dataset* (SS-Twitter) e *Sentiment Evaluation Datas*et (SE-Twitter).

Os resultados experimentais indicam que a remoção de URLs, a remoção de *stop words* e a remoção de números minimamente afetam o desempenho de classificadores. Além disso, substituir a negação e expandir as siglas pode melhorar a precisão da classificação. Portanto, a remoção de palavras, números e URLs é apropriada para reduzir o ruído, mas não afeta o desempenho. Substituir a negação é eficaz para a análise do sentimento.

Outro método híbrido interessante foi apresentado em [59], nesse trabalho foi combinado uma rede neural artificial dinâmica com *n-grams*. *Emoticons* e *tweets* que continham a palavra amor ou ódio ou seus sinônimos foram utilizados como recursos para construir os dois classificadores: SVM e uma Arquitetura Dinâmica para Redes Neurais Artificiais (DAN2). A abordagem proposta foi testada em uma coleção de *tweets* rastreados utilizando a entidade Justin Bieber. Os resultados mostraram que o DAN2 conseguiu superar a SVM.

Em [60] os autores desenvolveram um estudo comparativo dos classificadores mais comuns utilizados na literatura na tarefa de análise de sentimento. Nesse estudo os autores utilizaram os classificadores *Multinomial Naïve Bayes*, *Bernoulli Naïve Bayes* e SVM. Foi utilizado, também, o conjunto de *Stanford Twitter Sentiment Data* com dados coletados em 2009.

No pré-processamento todos os nomes de usuários do *Twitter* que iniciam com o símbolo @ foram substituídos pelo termo "USERNAME", todos os *links* de URL foram substituídos pelo termo "URL", redução do número de letras repetidas mais de duas vezes em todas as palavras (por exemplo, a palavra "haaaappy" ficaria "haappy" depois da redução), remoção de todas as *hashtags* do *Twitter* que começam com #, remoção de todos os *emoticons*

à medida que adicionam ruído durante o treinamento dos classificadores. Foram escolhidos *unigrams* e *bigrams* como atributos.

Os resultados experimentais mostraram que o classificador *Multinomial Naïve Bayes* superou outros classificadores examinados no estudo, sendo o menos afetado pela dispersão do conjunto de dados utilizado. *Unigrams,* como forma de representação dos dados, mostrou-se mais eficaz nesse contexto, pois produzem conjuntos de dados menos esparsos.

O *framework* proposto em [61] foi desenvolvido por um processo de três etapas, o último passo foi baseado em um método híbrido. A primeira etapa incluiu a aquisição de dados utilizando a API do *Twitter*, seguido de pré-processamento de *tweets*. O pré-processamento incluiu a detecção de gírias e abreviaturas, lematização, correção e remoção de *stop words*. Os *tweets* pré-processados foram, então, utilizados pelo classificador de sentimentos.

O PCA (*Polarity Classification Algorithm*) descrito pelo autor utilizou o EEC (*Enhanced Emoticon Classifier*), o IPC (*Improved Polarity Classifier*) e o SWNC (*Classificador SentiWordNet*). Um conjunto de *emoticons*, uma lista de palavras e o dicionário *SentiWordNet* foram utilizados pelos classificares EEC, IPC, e SWNC respectivamente. Os experimentos mostraram que a classificação híbrida final conseguiu superar o desempenho de usar qualquer um dos classificadores EEC, IPC ou SWNC separados.

A Tabela 2.5 resume os artigos que combinaram métodos de aprendizado de máquina e léxicos para abordar tarefa de análise de sentimento.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Jianqiang, Z. and Xiaolin, G. [58]	2017	SVM, *Naïve Bayes*, *Logistic Regression* e *Random Forest*	*unigrams*, *bigrams*, AFFIN4 e *SentiWordNet*	STS, SS-Twitter, SE-Twitter.
Ghiassi et al. [59]	2012	análise de n-gram, SVM, DAN2	*emoticons, tweets* contendo as palavras 'love' ou 'hate'.	próprio
Ismail, E., Harous, S. and Belkhouche, B. [60]	2016	Multinomial Naïve Bayes, Bernoulli Naïve Bayes e SVM.	*unigrams*, *bigrams*	STS
Khan et al. [61]	2014	EEC, IPC, SWNC	*emoticons*, palavras positivas e negativas, *SentiWordNet*.	próprio

Tabela 2.5: Sumário de artigos que empregaram métodos híbridos na tarefa de análise de sentimento.

2.4 Métodos Baseados em Grafos

Apesar dos métodos que utilizam aprendizado de máquina terem alcançado um bom desempenho na tarefa de análise de sentimento, tais métodos precisam de um grande número de dados marcados. A propagação de marcadores pode reduzir a demanda por dados marcados. Considerando essa situação, alguns pesquisadores utilizaram o gráfico social do *Twitter* supondo que pessoas influenciam umas às outras. A propagação de marcadores é um método semi-supervisionado no qual os rótulos são distribuídos aos nós utilizando gráficos de conexão. Tal abordagem não tem sido muito explorada recentemente, razão pela qual nossas referências serem relativamente antigas.

Em [62] os autores aplicaram métodos de propagação de rótulo na tarefa de análise de sentimento. O método proposto utilizou o gráfico de seguidores do *Twitter* sob a suposição

de que as pessoas influenciam umas às outras. Usuários, *tweets*, *unigrams*, *bigrams*, *hashtags* e *emoticons* foram utilizados como nós para a construção do grafo. O método de propagação de rótulo proposto superou uma abordagem baseada em léxico e um classificador *MaxEnt* (Máxima Entropia).

Em [63] os autores abordaram a tarefa de análise de sentimento com um método de propagação de rótulos baseado em análises de *tokens* de emoção. Os autores primeiramente extraíram os *tokens* de emoção dos *tweets*. Um método de propagação em grafos foi então utilizado para atribuir polaridades aos *tokens*. Na última etapa, eles analisaram e classificaram os *tokens*. Os *tokens* de emoção incluem *emoticons*, repetição de pontuação e repetição de letras.

Em [64] os autores propuseram um modelo baseado em grafos que utilizou a ocorrência concomitante de *hashtags* para classificar o sentimento de outras *hashtags*. Eles propuseram algoritmos diferentes (*Loopy Belief Propagation*, *Relaxation Labelling*, *Iterative Classification Algorithm*) que foram comparados com o algoritmo de SVM (*Support Vector Machine*). O SVM foi treinado com vários atributos, incluindo *unigrams*, pontuação e *emoticons*. O algoritmo *Loopy Belief Propagation* conseguiu obter o melhor desempenho em termos de precisão em comparação com os outros métodos testados.

Em [65] os autores utilizaram as relações sociais dos usuários para abordar a tarefa de análise de sentimento no nível de usuário. Seu estudo mostrou que os usuários conectados compartilham o mesmo sentimento. Além disso, eles empiricamente provaram que se dois usuários compartilham a mesma opinião, então eles têm maior probabilidade de ter uma conexão em uma rede social.

Os autores compararam três métodos: SVM Vote, *Heterogeneous Graph Model* com estimativa direta a partir de estatísticas simples (*HGM-NoLearning*) e *Heterogeneous Graph Model* com *SampleRank* (*HGM-Learning*). Os autores avaliaram seu método em *tweets* sobre políticos e mostraram que a tarefa de análise de sentimento em nível de usuário poderia ser significativamente melhorada ao considerar as conexões dos usuários dentro de uma rede social.

A Tabela 2.6 resume os artigos que empregaram métodos baseados em grafos e que tratam o problema de análise de sentimento.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Speriosu et al. [62]	2011	*LexRatio*, *MaxEnt*, *LProp*.	*n-gram, hashtags, emoticons, léxicos, Twitter follower graph.*	STS, OMD, HCR.
Cui et al. [63]	2011	*graph propagation.*	Emoticons, pontuação, *SentiWordNet*.	STS.
Wang et al. [64]	2011	SVM-voting, *Loopy Belief Propagation, Relaxation Labeling, Iterative Classification Algorithm*.	*unigrams*, pontuação, *emoticons*, léxicos.	Próprio
Tan et al. [65]	2011	SVM Vote, HGM-*NoLearning*, *HGM-Learning.*	Seguidores e seguidos	Próprio

Tabela 2.6: Sumário de artigos que empregaram métodos de grafos na tarefa de análise de sentimento.

2.5 Outros Métodos

Na literatura existem algumas técnicas que não podem ser categorizadas em nenhuma das categorias acima e têm sido pouco exploradas. Análise Formal de Conceitos, proposta em [66], é uma dessas técnicas. Nesse trabalho foi utilizado uma análise de conceito para construir um modelo de domínio ontológico. Os autores propuseram um método no qual *tweets* foram divididos em um conjunto de atributos que eram relevantes para o assunto. Seu

modelo foi aplicado e avaliado no domínio de *smart phones*. Considerando que o modelo detectou atributos do domínio e atribuiu pontuações a eles, eles conseguiram obter uma análise mais detalhada dos sentimentos em relação a um tópico específico.

Em [67] os autores utilizaram teoria de avaliação para determinar o sentimento da entidade principal de um *tweet*. Foi criado um dicionário de avaliação com uma lista de termos. A abordagem proposta foi avaliada no conjunto de dados de *Sanders* e superou o *baseline* [51], obtendo uma precisão de 87,57%.

Outra abordagem que se enquadra nessa categoria é a abordagem sociológica no manuseio de textos ruidosos e curtos, proposta em [68]. Essa abordagem foi baseada nas características do *Twitter* como dados conectados. Em particular, os autores apresentaram um método que incorporou teorias de consistência de sentimento e contagio emocional no processo de aprendizagem supervisionada. Os resultados experimentais mostraram que estas teorias sociais eram eficazes para na tarefa de análise de sentimento.

A Tabela 2.7 resume os artigos que empregaram outros métodos utilizados para abordar a tarefa de análise de sentimento.

Estudo	Ano	Algoritmos	Atributos	Conjunto de dados
Kontopoulos et al. [66]	2013	FCA	*WordNet*, *OpenDover*	próprio
Korenek and Simko [67]	2014	Teoria de avaliação e SVM	Linguística e aspectos de avaliação	*Sanders*
Hu et al. [68]	2013	SANT	Recursos de avaliação, *unigrams*	STS, OMD

Tabela 2.7: Sumário de artigos que empregaram outros métodos na tarefa de análise de sentimento.

2.6 Métricas de Avaliação para Análise de Sentimento em *Twitter*

Antes de falar sobre métricas é conveniente relembrar alguns conceitos como Verdadeiro Positivo (VP), Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Negativo (VN). A tabela abaixo, chamada de matriz de confusão, resume o significado desses termos:

	previsto como positivo	previsto como negativo
é positivo	VP	FN
é negativo	FP	VN

Tabela 2.8. Exemplo de matriz de confusão.

Como mencionado anteriormente, a tarefa de análise de sentimento pode ser considerada como um problema de classificação, uma vez que o objetivo no cenário típico é classificar a opinião expressa em um *tweet* como positiva ou negativa. As métricas de avaliação mais utilizadas são acurácia, precisão, *recall* e *F-measure*, adotadas a partir de problemas de classificação tradicionais [32].

Baseado na Tabela 2.8 apresentamos as mais populares métricas de avaliação utilizadas na literatura:

- **Acurácia:** A acurácia é a métrica de avaliação mais utilizada e mede a frequência com que o método a ser avaliado fez a previsão correta. É calculada como a soma das previsões verdadeiras dividida pelo número total de previsões. Ou seja:

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

- **Precisão:** A precisão representa a exatidão do método e é calculada como a proporção de instâncias que foram previstas como positivas e foram de fato positivas divididas pelo número total de instâncias que foram previstas como positivas. Ou seja:

$$Precisão = \frac{VP}{VP + FP}$$

- ***Recall*:** *Recall* denota a fração de instâncias positivas que foram previstas como positivas. Ou seja:

$$Recall = \frac{VP}{VP + FN}$$

- ***F-measure*:** Geralmente, calcular o *recall* e a precisão não é suficiente. Uma combinação dos dois é mais apropriada para avaliar o desempenho dos métodos. O *F-measure* é uma métrica que combina *recall* e precisão. Esta métrica é

também conhecida como *F1-score*, *F-measure* harmônico, *F1-measure* ou acurácia *F-measure* e é calculada como:

$$F - measure = 2 . precisão. \frac{recall}{precisão + recall}$$

Quando a classificação de sentimentos é formulada como um problema de múltiplas classes como, por exemplo, classificar um *tweet* como positivo, negativo ou neutro é prática comum calcular o *F-measure* positivo, negativo e neutro.

Capítulo 3
Metodologia Proposta

Esse Capítulo contém uma descrição dos objetivos desse trabalho e a descrição dos dados coletados. Mostraremos, também, os pré-processamentos utilizados e seus respectivos subníveis; descreveremos, de forma intuitiva a técnica *Term-Frequency Inverse Document Frequency* (TF-IDF), apresentaremos as técnicas supervisionadas que foram utilizadas e o tipo de literatura que foi consultada.

Por fim, apresentaremos o método de avaliação, os protocolos de testes e os trabalhos que alcançaram o estado da arte nesse domínio utilizando as mesmas bases que foram utilizadas em nossos experimentos.

3.1 Objetivo

Esse trabalho tem como objetivo:

- Verificar a influência de algumas técnicas de pré-processamento na melhora da acurácia dos classificadores SVM, *Random Forest*, *Logistic Regression* e *Naïve Bayes* no contexto de análise de sentimento.

- Verificar a influência da técnica TF-IDF na melhora da acurácia dos classificadores SVM, *Random Forest*, *Logistic Regression* e *Naïve Bayes* no contexto de análise de sentimento.

- Verificar a influência do volume do conjunto de treinamento na melhora da acurácia dos classificadores SVM, *Random Forest*, *Logistic Regression* e *Naïve Bayes* no contexto de análise de sentimento.

- Analisar técnicas que utilizam *ensembles* de classificadores.

Vale ressaltar que a análise de sentimento em todas as situações será feita utilizando dois níveis de sentimento (positivo e negativo) no nível de sentença. Vale ressaltar também que, para a tarefa da análise de sentimentos em *tweets,* não há diferença fundamental entre o nível de documento e o nível de sentença [32].

3.2 Metodologia

3.2.1 Coleta de Dados Marcados

A bases de dados utilizadas nesse trabalho foram:

- **Stanford – Twitter Sentiment Corpus**: Esse conjunto de dados [51] possui 1.600.000 *tweets* coletados por uma API do *Twitter*. Essa API envia, periodicamente, uma requisição de busca para o *emoticon* - :) - e outra requisição de busca para o *emoticon* - :(- ao mesmo tempo. Após a remoção *retweets*, qualquer *tweet* contendo ambos *emoticons*, *tweets* repetidos obteve-se 800.000 *tweets* com *emoticons* positivos e 800.000 *tweets* com *emoticons* negativos. Ao contrário do conjunto de treinamento, que foi gerado com base em *emoticons* específicos, o conjunto de teste foi gerado utilizando-se de uma API do *Twitter* com consultas específicas que inclui nomes de produtos, empresas e pessoas. Os *tweets* foram marcados manualmente como positivo ou negativo gerando 177 *tweets* negativos e 182 *tweets* positivos no conjunto de teste.

- **Obama-McCain Debate (OMD):** Esse conjunto de dados possui 3238 *tweets* coletados durante o primeiro debate presidencial dos EUA na TV que aconteceu em setembro de 2008 [52]. A classificação do sentimento desses *tweets* foi feita utilizando o *Amazon Mechanical Turk E4*.
  Cada *tweet* foi avaliado como positivo, negativo, neutro e outros. "Outros" *tweets* são aqueles que não puderam ser classificados. Foram mantidos apenas os *tweets* avaliados por, pelo menos, três eleitores, que formaram um conjunto de 1906 *tweets*, onde 710 classificados como positivos e 1196 classificados como negativos.

- **Health Care Reform (HCR):** Esse conjunto de dados possui *tweets* que contêm a *hashtag* "#hcr" (*health care reform*) que foram coletados em março de 2010 [54]. Um subconjunto desses dados foi anotado manualmente como positivo, negativo e neutro. Como mencionado anteriormente, os *tweets* neutros não serão considerados nesse estudo, assim, a formação do conjunto de dados de treinamento contém 621 *tweets* (215 positivos e 406 negativos), enquanto o conjunto de teste contém 665 *tweets* (154 positivos e 511 negativos).

- **Grand Old Party Debate (OPD):** Essa base de dados possui *tweets* sobre o debate do partido republicano no inicio de agosto de 2015 em Ohio [69]. Essa base de dados possui um total de 13.868 *tweets*, sendo 8492 negativos, 3141 neutros e 2235 positivos.

3.2.2 Pré-processamento

3.2.2.1 Estratégia de Pré-processamento I

Nessa primeira estratégia utilizaremos para representar os *tweets* quatro formas diferentes de pré-processamento que chamaremos de subnível I, subnível II, subnível III e subnível IV. Utilizaremos os pré-processamentos utilizados em [70], [71], [72] e [30] para gerar os quatro subníveis de pré-processamento.

No pré-processamento do subnível I iremos considerar:

- Remoção de pontuação não relevante (todas as pontuações com exceção de interrogação e exclamação) [70].
- Conversão das letras de maiúscula para minúscula [70].
- Remoção de *hyperlinks* [70].
- Remoção de números [70].
- Remoção de artigos [70].
- Remoção de pronomes, preposições e conjunções [70].
- Lematização [70].

No pré-processamento do subnível II iremos, a partir da saída do pré-processamento subnível I, considerar:

- Identificação de pontuação relevante (exclamação e interrogação) [70].
- Estender abreviação (por exemplo - "*aren't*" ficaria "*are not*") [70].
- Identificação de palavras alongadas (palavras como *gooooood* e *niiiice* seriam substituídas por LONG_WORD) [70].
- Identificação de negação (palavras como *no*, *not* seriam substituídas por

NEGATION_WORD)

- Identificação de interjeição positiva/negativa [70]; utilizaremos para isso uma lista de interjeições positivas e negativas [73].

No processamento do subnível III iremos, a partir da saída do pré-processamento II, considerar:

- Identificação de *hashtags* (substituiremos as *hashtags* positivas por positive_hashtag e *hashtags* negativas por negative_hashtags) [71].
- Identificação de *emoticons* (substituiremos as *emoticons* positivos por positive_emoticons e *emoticons* negativos por negative_emoticons) [72].

No processamento do subnível IV iremos, a partir da saída do pré-processamento III, considerar:

- Identificação de palavras positivas e negativas [30] (substituiremos as palavras positivas por positive_word e as palavras negativas por negative_word).

Utilizando essa estratégia iremos, portanto, obter 4 representações dos *tweets*. Após obtermos as 4 representações de todos os *tweets* nos conjuntos de treinamento e teste utilizaremos o método de *bag-of-words (BoW)* para representar os *tweets* obtendo, portando, 4 matrizes de representação do conjunto de treinamento e 4 matrizes de representação no conjunto de teste.

3.2.2.2 Estratégia de Pré-processamento II

A estratégia de pré-processamento II teve como base [74]. Nessa estratégia primeiramente iremos gerar uma lista de *unigrams* e outra lista de *bigrams* utilizando o conjunto de treinamento. Para a geração das listas primeiramente iremos processar os *tweets* do conjunto de treinamento da seguinte forma:

- Remoção de URLs, *hashtags,* pontuação e números.
- Expansão de contrações (expandiríamos, por exemplo, *I'm* para *I am*).
- Conversão de letras para minúsculo.
- Correção ortográfica.
- Remoção de artigos, conjunções, preposições, pronomes.
- *Steemming.*

- Remoção de palavras com tamanho menor que 2.

Com os *tokens* restantes iremos criar uma lista com *unigrams* e *bigrams* com suas respectivas ocorrências em *tweets* positivos e negativos, por exemplo os *tweets*:

tweet 01 – word01 word02 word03; positive tweet
tweet 02 – word06 word07 word01; negative tweet
Teríamos a seguinte lista de unigrams:
unigram,positive.ranking,negative.ranking
word01,1,1
word02,1,0
word03,1,0
word06,0,1
word07,0,1

E a seguinte lista bigram:
bigram,positive.ranking,negative.ranking
word01 word02,1,0
word02 word03,1,0
word06 word07,0,1
word07 word01,0,1

Iremos utilizar as duas listas para classificar se um *unigram* ou *bigram* é positivo ou negativo baseado em seus *rankings*.

Após remover pontuação e números, expandir contrações, converter maiúsculo em minúsculo; fazer correção ortográfica, remover artigos, conjunções, preposições, pronomes, fazer *stemming* e remover palavras que possuem menos de dois caracteres. Iremos representar os *tweets* em forma de atributos, dividindo os mesmos em subníveis como feito na estratégia de pré-processamento I, da seguinte forma:

- Subnível I
    - Verificar presença de URL.
    - Verificar presença de *targets* (@).
- Subnível II
    - Verificar quantidade de *hashtags* positivas e negativas presentes no *tweet*.
    - Verificar quantidade de *emoticons* positivos e negativos presentes.

- Subnível III
    - Verificar presença de símbolos especiais (!).

- Subnível IV
    - Verificar quantidade de *unigrams* positivos e negativos presentes no *tweet*.
    - Verificar quantidade de *bigrams* positivos e negativos presentes no *tweet*.

3.2.3 *Term Frequency–Inverse Document Frequency* (TF-IDF)

Explicaremos, de maneira intuitiva, a técnica TF-IDF. Essencialmente, TF-IDF determina a frequência relativa de palavras em um documento comparada com a proporção inversa da palavra ao longo de todo o corpo do documento. Intuitivamente, esse cálculo determina o quão relevante uma determinada palavra é em um determinado documento.

As palavras que são comuns em uma única ou um pequeno grupo de documentos tendem a ter um numero maior de TF-IDF do que palavras comuns, tais como artigos e preposições. Aplicaremos tal técnica na estratégia de pré-processamento I pois a mesma utiliza o método *BoW* considerando todas as palavras de todos os *tweets*. Utilizar essa estratégia na estratégia de pré-processamento II não faz sentido devido ao reduzido numero de atributos.

3.2.4 Avaliação, Classificadores Supervisionados e Revisão da Literatura

Iremos utilizar a acurácia como métrica de avaliação por ser a mais utilizada na literatura [32] e ser suficiente para o objetivo desse trabalho. Utilizaremos as técnicas supervisionadas de *Support Vector Machines* (SVM), *Random Forest*, *Logistic Regression*, *Naïve Bayes* e *ensembles* dessas técnicas para treinar e classificar como positivo ou negativo os *tweets* das quatro bases de dados mencionadas. Tais classificadores foram escolhidos pois são usualmente utilizados como prova de conceito [38].

Nas técnicas de SVM e *Random Forest* iremos variar os hiperparâmetros. No classificador SVM iremos variar os hiperparâmetros alfa e custo além do *kernel* e no

classificador *Random Forest* iremos variar os hiperparâmetros *ntree* e *mtry*. Falaremos sobre os valores utilizados nos hiperparâmetros na seção 3.2.5 desse trabalho.

Iremos comparar a acurácia alcançada pelos nossos modelos, nas bases utilizadas, com trabalhos que alcançaram o estado da arte nas mesmas bases. Iremos descrever esses trabalhos na seção 3.2.6 desse trabalho.

Os trabalhos que utilizaremos como revisão de literatura estão relacionados com técnicas de pré-processamento de textos, escolha de atributos em textos e utilização de classificadores supervisionados para classificar sentimentos em textos.

3.2.5 Protocolo de Testes

Nessa seção faremos algumas considerações sobre o protocolo de teste utilizados:

- **Sobre as tecnologias utilizadas:** Os experimentos utilizaram na execução as instâncias *EC2 c4.4xlarge* (16 vCPU, 62 EPU, 16 GiB de memória) da *cloud* da AWS [75] e a API *dashDb* da *cloud bluemix* versão *free* (8 GB RAM, 2 cores, 500 GB em disco) [76]. Nos experimentos utilizando a AWS foram utilizadas as imagens disponibilizadas em [77]. Utilizamos extensivamente a API *dashDb* para executar os pré-processamentos e a AWS para executar os testes com classificadores supervisionados.

- **Sobre a linguagem de programação e as bibliotecas utilizadas:** Utilizamos a linguagem R juntamente com as bibliotecas 'e1071' [78] para os classificadores SVM, *Random Forest* e *Naïve Bayes* e a biblioteca LOGIT [79] para o classificador *Logistic Regression*.

- **Sobre o pré-processamento:** As quatro representações de *tweets* em cada uma das duas estratégias de pré-processamento foram criadas de forma incremental, ou seja, primeiro foi criada a representação utilizando os atributos do subnível I, a saída da representação gerada por tal subnível serviu de entrada para que fosse gerada a representação do subnível II e assim sucessivamente para os subníveis III e IV.

- **Sobre os hiperparâmetros utilizados no classificador SVM:** Os hiperparâmetros custo e alfa foram utilizados com valores entre 2^{-8} e 2^{8} (a variação foi feita dobrando-se os valores, por exemplo, primeiro se utilizou 2^{-8}, depois 2^{-7} e assim, sucessivamente). Foram utilizados os *kernels linear* e *radial*. Variamos esses hiperparâmetros quando comparamos as técnicas de pré-processamento TF (*Term Frequenc*y) e TF-IDF (*Term Frequency-Inverse Document Frequency*) e quando avaliamos técnicas *ensembles*. Utilizamos o melhor valor alcançado na acurácia com essa variação de hiperparâmetros.

- **Sobre os hiperparâmetros utilizados no classificador *Random Fore*st:** Nesse classificador os hiperparâmetros *ntree* e *mtry* foram alterados. Os valores de *ntree* utilizados foram 500, 1000 e 1500 e variamos os valores de *mtry* da seguinte forma:

$$x * \sqrt{< numero\ de\ atributos >} + 1; \tag{3.1}$$

$$x * log2 < numero\ de\ atributos >; \tag{3.2}$$
$$sendo\ x = 1.5,\ 2.0,\ 2.5,\ 3.5$$

Variamos esses hiperparâmetros quando comparamos as técnicas de pré-processamento TF (*Term Frequenc*y) e TF-IDF (*TermFrequency-Inverse Document Frequency*) e quando avaliamos técnicas *ensembles*. Iremos utilizar o melhor resultado alcançado na acurácia com essa variação de hiperparâmetros.

- **Sobre os testes:** Para as bases de dados que não possuíam *tweets* para treinamento e teste separados (bases de OMD e OPD) foi utilizado a técnica de *holdout* [80]. Nessa técnica os conjuntos de treinamento e teste foram divididos aleatoriamente na proporção de 2/3 dos dados para treinamento e 1/3 para teste. Para cada vez em que cada algoritmo de classificação foi executado foram gerados 5 conjuntos de treinamento e teste. A criação de tais conjuntos foi feita aleatoriamente e utilizamos o melhor resultado alcançado na acurácia com essa variação de conjuntos.

- **Sobre as técnicas *ensembles*:** Para criar os *ensembles* de classificadores iremos

utilizar uma combinação de 3 classificadores, em cada uma das estratégias de pré-processamento. A escolha dos classificadores será baseada nas 3 melhores acurácias individuais. Iremos utilizar votação majoritária para que esses classificadores decidam sobre a natureza positiva ou negativa do *tweet*.

3.2.6 Estado da Arte

Como mencionado anteriormente iremos comparar a acurácia alcançada pelos nossos modelos, nas bases utilizadas, com trabalhos que alcançaram o estado da arte nas mesmas bases. Seguem as bases de dados e os trabalhos que alcançaram a melhor acurácia que encontramos na literatura nessas bases:

- **Base de Stanford**: Em [74] os autores utilizaram os atributos de *unigram*, *bigram*, *hashtag*, *emoticons*, URLs, *targets*, e símbolos especiais juntamente com o classificador SVM. O trabalho conseguiu 87.20% de acurácia na classificação.

- **Base HCR**: Em [81] os autores utilizaram uma combinação de DeepCNN (*Deep Convolutional Neural Networks*) e Bi-LSTM (*Bi-Long Short Term Memory networks*) para endereçar o problema. Esse modelo conseguiu 80.90% de acurácia na classificação.

- **Base OMD**: Em [82] os autores propuseram um método de classificação de polaridade baseado em similaridade de *tweets* utilizando três passos: Primeiramente os n *tweets* mais similares ao *tweet* que se quer classificar são selecionados; após isso, um conjunto de atributos são escolhidos e, finalmente, classificadores supervisionados são utilizados para classificar o *tweet*. Na base OMD esse método alcançou 77.50% de acurácia.

- **Base OPD:** Em [83] o autor cita um trabalho feito na competição *Kaggle* [84] e foi o melhor resultado que encontramos para essa base. O autor do trabalho utilizou LSTM (*Long Short Term Memory networks*) e conseguiu alcançar 84.00% de acurácia na classificação utilizando essa base.

Capítulo 4
Testes, Avaliação e Discussão de Resultados

Nesse Capítulo iremos avaliar a influência das estratégias de pré-processamento I e II e seus respectivos subníveis. Avaliaremos os ganhos adquiridos em cada subnível e o ganho total. Discutiremos, também, a Influência da técnica TF-IDF, do volume do conjunto de treinamento e avaliaremos técnicas *ensembles*. Utilizamos acurácia como métrica de desempenho dos classificadores avaliados.

Como mencionado em detalhes na seção em que descrevemos o nosso protocolo de testes (seção 3.2.5), variamos hiperparâmetros e utilizamos os melhores resultados dessa variação nos classificadores SVM e *Random Forest* quando comparamos as técnicas de pré-processamento TF (*Term Frequenc*y) e TF-IDF (*Term Frequency-Inverse Document Frequency*) e avaliarmos técnicas *ensembles*.

4.1 Influência do Pré-processamento no Desempenho de Classificadores Supervisionados

Como mencionado anteriormente, nessa seção, analisaremos a influência do pré-processamento nos classificadores supervisionados SVM, *Logistic Regression*, *Random Forest* e *Naïve Bayes*, os quais foram treinados e testados utilizando as estratégias de pré-processamento I e II com seus respectivos subníveis. Foram utilizadas, nesses experimentos, as bases de dados de *Stanford*, OMD (*Obama-McCain Debate*), HCR (*Health Care Reform*) e OPD (*First Old Party Debate*).

Foram realizados quatro experimentos utilizando os classificadores supervisionados SVM, *Random Forest*, *Logistic Regression* e *Naïve Bayes* os quais descreveremos e discutiremos nas próximas seções.

4.1.1 Experimento 01 – Influência do Pré-processamento no Classificador Supervisionado SVM

Nesse experimento foi utilizado o classificador SVM nas bases de dados citadas e pré-processadas utilizando as estratégias de pré-processamento I e II.

Na Tabela 4.1 observamos os resultados dos pré-processamentos I e II com seus respectivos subníveis. Nos pré-processamentos I e II o classificador foi treinado, no conjunto de *Stanford,* com 3000 *tweets*. Destacamos, em negrito, os melhores resultados obtidos em cada base de dados, em cada um dos pré-processamentos.

	Acurácia obtida em cada subnível da estratégia de pré-processamento I (%)				**Acurácia obtida em cada subnível da estratégia de pré-processamento II (%)**			
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**
Stanford	75.76	76.04	76.32	**77.99**	54.87	55.71	56.26	**68.24**
OMD	73.20	73.50	73.80	**73.66**	52.37	54.67	60.03	**67.07**
HCR	74.43	75.18	75.78	**76.39**	57.89	60.00	65.44	62.98
OPD	75.02	75.15	**76.97**	76.01	62.98	66.99	67.98	**71.98**

Tabela 4.1: Acurácia obtida utilizando as estratégias de pré-processamento I, II e o classificador SVM.

4.1.2 Experimento 02 – Influência do Pré-processamento no Classificador Supervisionado *Random Forest*

Nesse experimento foi utilizado o classificador *Random Forest* nas bases de dados citadas e pré-processadas utilizando as estratégias de pré-processamento I e II.

Na Tabela 4.2 observamos os resultados dos pré-processamentos I e II com seus respectivos subníveis. Nos pré-processamentos I e II o classificador foi treinado, no conjunto de *Stanford*, com 3000 *tweets*. Destacamos, em negrito, os melhores resultados obtidos em cada base de dados, em cada um dos pré-processamentos.

	Acurácia obtida em cada subnível da estratégia de pré-processamento I (%)				**Acurácia obtida em cada subnível da estratégia de pré-processamento II (%)**			
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**
Stanford	71.03	73.53	72.98	**81.95**	55.15	64.90	59.88	**72.98**
OMD	70.59	71.97	73.50	**75.34**	54.67	52.37	66.92	**69.98**
HCR	69.92	69.92	71.12	**76.09**	58.94	56.99	60.00	**69.92**
OPD	71.01	72.01	73.02	**78.04**	53.03	55.01	62.03	**71.03**

Tabela 4.2: Acurácia obtida utilizando as estratégias de pré-processamento I, II e o classificador *Random Forest*.

4.1.3 Experimento 03 – Influência do Pré-processamento no Classificador Supervisionado *Logistic Regression*

Nesse experimento foi utilizado o classificador *Logistic Regression* nas bases de dados citadas e pré-processadas utilizando as estratégias de pré-processamento I e II.

Na Tabela 4.3 observamos os resultados do pré-processamento I e II com seus respectivos subníveis. Nos pré-processamentos I e II o classificador foi treinado, no conjunto de *Stanford*, com 3000 *tweets*. Destacamos, em negrito, os melhores resultados obtidos em cada base de dados, em cada um dos pré-processamentos.

	Acurácia obtida em cada subnível da estratégia de pré-processamento I (%)				**Acurácia obtida em cada subnível da estratégia de pré-processamento II (%)**			
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**
Stanford	55.98	62.11	62.11	**78.27**	64.90	67.96	70.75	**72.98**
OMD	55.89	59.41	65.39	**73.04**	52.98	54.88	59.11	**63.89**
HCR	55.63	62.85	65.26	**76.24**	53.08	55.63	57.89	**68.87**
OPD	61.04	65.02	70.01	**79.06**	54.98	59.03	63.05	**72.03**

Tabela 4.3: Acurácia obtida utilizando as estratégias de pré-processamento I, II e o classificador *Logistic Regression*.

4.1.4 Experimento 04 – Influência do Pré-processamento no Classificador Supervisionado *Naïve Bayes*

Nesse experimento foi utilizado o classificador *Naïve Bayes* nas bases de dados citadas e pré-processadas utilizando as estratégias de pré-processamento I e II.

Na Tabela 4.4 observamos os resultados do pré-processamento I e II com seus respectivos subníveis. Nos pré-processamentos I e II o classificador foi treinado, no conjunto de *Stanford*, com 3000 *tweets*. Destacamos, em negrito, os melhores resultados obtidos em cada base de dados, em cada um dos pré-processamentos.

	Acurácia obtida em cada subnível da estratégia de pré-processamento I (%)				**Acurácia obtida em cada subnível da estratégia de pré-processamento II (%)**			
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**
Stanford	**49.30**	**49.30**	**49.30**	**49.30**	55.15	55.98	58.77	**74.73**
OMD	54.88	63.88	67.98	**69.98**	54.67	55.89	70.59	**77.92**
HCR	49.62	51.12	54.88	**60.15**	55.63	57.89	68.87	**76.09**
OPD	55.05	58.12	62.94	**65.02**	57.08	55.15	64.04	**78.02**

Tabela 4.4: Acurácia obtida utilizando as estratégias de pré-processamento I, II e o classificador *Naïve Bayes*.

4.1.5 Discussão de Resultados da Influência do Pré-processamento no Desempenho de Classificadores Supervisionados

Nessa seção analisaremos a influência da adição de diferentes tipos de pré-processamentos na acurácia de classificadores supervisionados no contexto de análise de sentimentos em *tweets*.

A Tabela 4.5 mostra o ganho na acurácia média dos classificadores supervisionados, em pontos percentuais, nas bases utilizadas quando adicionamos, de forma incremental, diferentes subníveis de pré-processamento. Com base nos dados da tabela a primeira conclusão que chegamos é que, na média, sempre quando adicionamos pré-processamento temos uma melhora na acurácia. Observando os ganhos abaixo temos que o ganho total foi, no pior caso, de 1.39 pontos percentuais e, no melhor caso, de 22.30 pontos percentuais.

Diferentes estudos constataram que adicionar pré-processamento contribui para melhoria do desempenho de classificadores supervisionados como, por exemplo, os estudos descritos em [85] [86] [87]; os motivos vão desde diminuição no ruído dos dados [88], passando por diminuição de dados esparsos [89], seleção de melhores atributos [90] entre outros.

	estratégia de pré-processamento I				**estratégia de pré-processamento II**			
	sub nível II (%)	**sub nível III (%)**	**sub nível IV (%)**	**ganho total (%)**	**sub nível II (%)**	**sub nível III (%)**	**sub nível IV (%)**	**ganho total (%)**
SVM	0.36	0.74	0.29	1.39	3.82	3.02	8.48	15.32
Random Forest	1.21	0.79	4.97	5.31	1.88	6.13	9.02	17.03
Logistic Regression	5.21	3.34	10.95	19.50	2.92	3.44	7.51	13.87
Naïve Bayes	3.39	3.17	2.33	8.89	1.88	9.47	10.95	22.30
Ganho Médio	2.54	2.01	4.63	8.77	2.62	5.51	8.98	17.13

Tabela 4.5: Tabela de ganhos na acurácia em cada subnível na estratégia de pré-processamento I e II por classificador.

Analisando os resultados da acurácia nos classificadores separadamente, observamos que, utilizando a estratégia de pré-processamento I, o classificador *Logistic Regression* responde melhor (ganho total na acurácia de 19.50 pontos percentuais) e o classificador SVM responde pior (1.39 pontos percentuais). Utilizando a estratégia de pré-processamento II o classificador *Naïve Bayes* responde melhor (ganho total na acurácia de 22.30 pontos percentuais) e o classificador *Logistic Regression*, pior (ganho total na acurácia de 13.87 pontos percentuais).

A diferença fundamental entre os dois tipos de pré-processamentos está relacionada com as matrizes de dados geradas. No pré-processamento I as matrizes geradas são mais esparsas que as matrizes geradas no pré-processamento II. Isso nos faz concluir que diferentes classificadores possuem diferentes desempenhos conforme os dados sejam mais ou menos esparsos. Essa conclusão coincide com os resultados dos estudos [91] [92] [93] [94] [95].

Em [91] os autores mencionam a dificuldade do classificador SVM em definir os hiperplanos na fase de treinamento utilizando dados esparsos, em [92] os autores concluíram que, apesar do SVM com o *kernel linear* (utilizado nesse experimento) conseguir desempenho

eficiente na classificação de dados esparsos, esse desempenho pode ter taxa de erro aumentada utilizando dados esparsos.

Em [93] os autores mencionam as dificuldades em se utilizar o classificador *Naïve Bayes* em problemas de classificação com dados esparsos pelo fato desse classificador precisar estimar diferentes parâmetros. Em [94] os autores utilizaram o classificador *Random Forest* para lidar com dados de pacientes que sofriam da doença *Alzheimer* (dados esses extremamente esparsos) e obtiveram ótimos resultados.

Os autores de [95] concluíram que, para dados aleatórios e esparsos, o classificador *Logistic Regression* deve ser considerado como primeira opção.

Pudemos constatar que o conjunto de atributos que mais contribuiu para a melhoria da acurácia é o conjunto de *unigrams* e/ou *bigrams* positivos e negativos. No pré-processamento I isso foi feito utilizando uma lista de palavras (apenas foi utilizado *unigrams*) e no pré-processamento II isso foi feito utilizando as palavras do conjunto de treinamento (*unigrams* e *bigrams*). Na média dos classificadores, tais atributos conseguiram uma melhoria de 4.63 e 8.98 pontos percentuais nas estratégias de pré-processamento I e II respectivamente.

Além de contribuir mais na média, esse conjunto de atributos contribuiu mais em 2 dos 4 casos nos classificadores que utilizaram a estratégia de pré-processamento I e em todos os casos nos classificadores que utilizaram a estratégia de pré-processamento II. No caso do pré-processamento I esse comportamento pode ser explicado pelo fato do *Naïve Bayes* ter um rendimento muito baixo quando utilizamos o pré-processamento I e no caso do SVM pelo fato de todos os pré-processamentos terem contribuído muito pouco para a melhoria na acurácia geral.

Nossos resultados parecem confirmar os resultados de outros estudos, em [74] os atributos que mais contribuíram para melhoria na acurácia foram as listas de *bigram* e *unigram* geradas com base no conjunto de treinamento quando utilizado o classificador SVM; em [51] e [96] foram reportados resultados compatíveis com o estado da arte quando utilizados *unigram* juntamente com algum classificador supervisionado.

Em [97] o autor utilizou *unigrams* e *bigrams* afirmando que ambos contribuem significativamente para melhora em problemas de classificação, com *unigrams* contribuindo de forma mais relevante, o autor atribui isso ao fato das listas de *bigrams* geradas serem extremamente esparsas.

Pode-se afirmar que, intuitivamente, é claro que *unigrams* contribuam de forma mais relevante para melhoria da acurácia, pois as palavras com conotação positiva/negativa são as que nos dão pistas mais claras do tipo de sentimento expresso em um determinado texto. Por exemplo, na frase "This paper is so nice", se isolarmos apenas a palavra "nice" a mesma já nos daria uma pista muito relevante do tipo de sentimento expresso no texto.

Lista de *bigrams* são muito úteis em situações onde aparecem negações. Por exemplo, na frase "This place is not good for party", apesar da palavra "good" ter uma conotação positiva a palavra "not" faz com que a conotação da ideia transmitida pela frase mude completamente.

Utilizando pré-processamento I e II o conjunto de atributos que menos contribuiu para a acurácia geral em todos os casos foi o conjunto de atributos do subnível III (identificação de *hashtags* e *emoticons*), isso está diretamente relacionado ao fato desses atributos serem pouco frequentes em *tweets* (estão presentes em menos de 10% dos *tweets*) [98].

4.2 Influência do Volume do Conjunto de Treinamento na Acurácia de Classificadores Supervisionados

Nessa seção analisaremos a Influência do volume do conjunto de treinamento na acurácia de classificadores supervisionados. Utilizaremos a estratégia de pré-processamento II e o conjunto de *Stanford* nesse experimento.

Utilizaremos essa estratégia de pré-processamento II com seu subnível IV pelo fato da mesma nos permitir aumentar substantivamente o conjunto de treinamento por não produzir representações de *tweets* com alta dimensionalidade.

Representações de *tweets* com alta dimensionalidade inviabilizaram experimentos com conjunto de treinamento muito grande pelo fato de não possuirmos capacidade computacional para tal, mesmo utilizando duas *clouds* com alto poder de processamento nos experimentos, como detalhado na nossa seção de protocolo de testes (seção 3.2.5). O subnível IV foi escolhido pelo fato desse conjunto de atributos alcançarem melhores resultados nos experimentos anteriores.

Utilizaremos o conjunto de dados de *Stanford*, pois se trata de um conjunto de dados bastante utilizado na literatura e possui um conjunto de treinamento maior que os de outras bases utilizadas nesse trabalho. Nos experimentos dessa seção treinamos o conjunto de *Stanford* com 3000 e 1.000.000 de exemplos.

4.2.1 Experimento e Discussão da Influência do Volume do Conjunto de Treinamento na Acurácia de Classificadores Supervisionados

A tabela 4.6 mostra a acurácia quando treinamos o conjunto de *Stanford* com 3000 e 1.000.000 de *tweets* e o ganho na acurácia em pontos percentuais.

	Acurácia utilizando a estratégia de pré-processamento II – *Stanford* com 3.000 *tweets* no treinamento em %	**Acurácia utilizando a estratégia de pré-processamento II - *Stanford* com 1.000.000 *tweets* no treinamento em %**	**Ganho na acurácia em pontos percentuais**
SVM	68.24	85.23	16.99
Random Forest	72.98	79.94	6.96
Logistic Regression	72.98	85.51	12.53
Naïve Bayes	74.73	80.22	5.49

Tabela 4.6: Tabela de acurácia e ganhos de acurácia em pontos percentuais quando variamos o volume do conjunto de treinamento utilizando a estratégia de pré-processamento II.

Observando os dados da tabela podemos observar ganhos significativos na acurácia quando aumentamos a quantidade de dados no conjunto de treinamento, conseguindo uma melhora na acurácia quando se aumenta o conjunto de dados de 3.000 para 1.000.000 de 16.99 pontos percentuais (85.23-68.24) no melhor caso (SVM) e de 5.49 (80.22-74.73) pontos percentuais no pior caso (*Naïve Bayes*).

Nossos resultados coincidem com os resultados encontrados em [98]. Esse trabalho utiliza (entre outros classificadores) os mesmos classificadores dessa dissertação em problemas de análise de sentimentos (entre outros de processamento de linguagem natural).

Os autores concluíram que o classificador SVM é o classificador que mais tem o seu desempenho afetado positivamente por variações progressivas do volume do conjunto de treinamento. Afirmaram, também, que tal classificador é o mais indicado em conjuntos com bases de dados grandes. Os autores constataram que o classificador *Naïve Bayes* é o classificador que menos aumenta positivamente seu desempenho quando aumentado o conjunto de treinamento.

O autor do trabalho em questão menciona que a natureza do classificador *Naïve Bayes* estaria relacionada com tal comportamento, já que tal classificador precisa de poucos dados de treinamento para definir parâmetros necessários para classificação e o aumento dos dados não influenciaria em definições de tais parâmetros.

Considerando nossos resultados e os resultados apresentados em [98] é correto afirmar que o classificador SVM produz hiperplanos que separa, de forma mais eficaz, as classes na medida em que o conjunto de treinamento aumenta. Concluímos, portanto, que a quantidade de volume do treinamento está diretamente relacionada com o aumento do desempenho de algoritmos supervisionados, porém, a quantidade de ganhos com o aumento do volume não é semelhante e está diretamente relacionada com a natureza do algoritmo supervisionado utilizado.

4.3 Influência da Técnica TF-IDF na Acurácia de Classificadores Supervisionados

Com o intuito de melhorar a acurácia do nosso modelo utilizaremos a técnica TF-IDF juntamente com a estratégia de pré-processamento I e iremos comparar os resultados com a técnica TF. Como mencionado em detalhes na seção que descrevemos nosso protocolo de testes (seção 3.2.5), no classificador SVM variamos o *kernel* e os hiperparâmetros alfa e custo e, no classificador *Random Forest* variamos os hiperparâmetros *ntree* e *mtry.* Reportaremos o melhor resultado da acurácia de tal variação.

Utilizaremos o pré-processamento I com subnível IV. Foi escolhido o subnível IV pelo fato de ter alcançado, na média dos 4 bancos de dados utilizados, nos experimentos anteriores, a melhor acurácia. Nessa seção não utilizaremos a técnica TF-IDF juntamente com

o pré-processamento II pelo fato desse tipo de pré-processamento ter um numero reduzido de atributos o que faz com que tal técnica não seja apropriada. Vale mencionar que o conjunto de *Stanford* foi treinado com 3000 *tweets*.

4.3.1 Experimento 01 – Influência da Técnica TF-IDF na Acurácia do Classificador SVM

Na Tabela 4.7 são apresentados os resultados da acurácia do classificador SVM utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV) nas bases de dados de *Stanford*, OMD, HCR e OPD. Apresentamos, também, o *kernel* e os hiperparâmetros utilizados.

base de *Stanford*				
kernel	***hiperparâmetro custo***	***hiperparâmetro alfa***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
radial	2^{8}	2^{-7}	80.05	**81.61**
base OMD				
kernel	***hiperparâmetro custo***	***hiperparâmetro alfa***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
radial	2^{7}	2^{-4}	74.11	**74.88**
base HCR				
kernel	***hiperparâmetro custo***	***hiperparâmetro alfa***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
radial	2^{7}	2^{-7}	76.39	**76.54**
base OPD				
kernel	***hiperparâmetro custo***	***hiperparâmetro alfa***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
radial	2^{7}	2^{-5}	78.02	**78.19**

Tabela 4.7: Resultados da acurácia do classificador SVM variando *kernel* e hiperparâmetros - estratégia de pré-processamento II (subnível IV) e técnicas TF e TF-IDF.

4.3.2 Experimento 02 – Influência da Técnica TF-IDF na Acurácia do Classificador *Random Forest*

Na Tabela 4.8 são apresentados os resultados da acurácia do classificador *Random Forest* utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV) nas bases de dados de *Stanford*, OMD, HCR e OPD. Apresentamos, também, os hiperparâmetros utilizados.

base de *Stanford*			
hiperparâmetro ntree	***hiperparâmetro k***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
1500	38.12	81.61	**81.89**
base OMD			
hiperparâmetro ntree	***hiperparâmetro k***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
1500	114.36	76.41	**76.87**
base HCR			
hiperparâmetro ntree	***hiperparâmetro k***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
1500	19.75	76.84	**77.29**
base OPD			
hiperparâmetro ntree	***hiperparâmetro k***	***Acurácia utilizando tf em %***	***Acurácia utilizando tf-idf em %***
1500	114.75	78.08	**79.12**

Tabela 4.8: Resultados da acurácia do classificador *Random Forest* variando hiperparâmetros - estratégia de pré-processamento II (subnível IV) e técnicas TF e TF-IDF.

4.3.3 Experimento 03 – Influência da Técnica TF-IDF na Acurácia do Classificador *Logistic Regression*

Na Tabela 4.9 são apresentados os resultados da acurácia do classificador *Logistic Regression* utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV) nas bases de dados de *Stanford*, OMD, HCR e OPD.

base de Stanford	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
78.97	**79.94**
base de OMD	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
73.04	**75.95**
base de HCR	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
76.24	75.18
base de OPD	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
79.06	77.04

Tabela 4.9: Resultados da acurácia do classificador *Logistic Regression* utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV).

4.3.4 Experimento 04 – Influência da Técnica TF-IDF na Acurácia do Classificador *Naïve Bayes*

Na Tabela 4.10 são apresentados os resultados da acurácia do classificador *Naïve Bayes* utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV) nas bases de dados de *Stanford*, OMD, HCR e OPD.

base de Stanford	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
49.3	**49.3**
base de OMD	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
69.98	**71.20**
base de HCR	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
60.15	59.54
base de OPD	
Acurácia utilizando tf em %	***Acurácia utilizando tf-idf em %***
65.02	**66.06**

Tabela 4.10: Resultados da acurácia do classificador *Naïve Bayes* utilizando as técnicas TF e TF-IDF juntamente com a estratégia de pré-processamento I (subnível IV).

4.3.5 Discussão da Influência da Técnica TF-IDF na Acurácia de Classificadores Supervisionados

Nessa seção discutiremos a influência da técnica TF-IDF na acurácia de classificadores supervisionados no contexto de análise de sentimentos em *tweets*.

A tabela 4.11 detalha os ganhos de acurácia média, dos classificadores supervisionados, em pontos percentuais, nas bases utilizadas quando utilizamos a técnica TF-IDF (apenas no pré-processamento nível I). Vale ressaltar que os dados da tabela 4.11 tiveram como base os resultados obtidos nas tabelas 4.7, 4.8, 4.9 e 4.10, ou seja, os dados foram obtidos da diferença entre os ganhos médios obtidos utilizando a técnica TF-IDF e a técnica TF.

SVM	
	Ganho médio adicionando a técnica TF-IDF
Ganho médio	1.78
Random Forest	
	Ganho médio adicionando a técnica TF-IDF
Ganho médio	1.71
Logistic Regression	
	Ganho médio adicionando a técnica TF-IDF
Ganho médio	0.20
Naïve Bayes	
	Ganho médio adicionando a técnica TF-IDF
Ganho médio	0.41

Tabela 4.11: Tabela de ganhos na acurácia utilizando a técnica de TF-IDF no pré-processamento nível I.

Analisando os ganhos nos classificadores individualmente (Tabela 4.11) não observamos uma melhoria significativa utilizando as técnicas TF-IDF; no pior caso conseguimos uma melhora de 0.20 pontos percentuais (classificador *Logistic Regression*) e no melhor caso uma melhora de 1.78 pontos (classificador SVM). Nosso resultado parece coincidir com o resultado de [99] e [100] onde os autores não conseguiram melhora expressiva adicionando a técnica TF-IDF.

No nosso estudo podemos explicar que esse ganho foi ainda menor pelo fato de já termos conseguido eliminar muito ruído utilizando pré-processamento. Como mencionado em [58] apesar de não contribuir para melhoria na acurácia, o subnível I do pré-processamento I é útil para que se diminua ruído, por exemplo, nesse subnível palavras como artigos que são muito comuns em textos já tinham sido removidas antes da utilização da técnica TF-IDF.

Caso esses atributos muito comuns estivessem sendo considerados quando a técnica TF-IDF foi aplicada a técnica apenas atribuiria um numero mais baixo para esses atributos. Isso explica o motivo pelo qual essa técnica não ter conseguido ganhos significativos.

4.4 Experimento e Discussão da Influência de Técnicas *Ensembles* na Acurácia de Classificadores Supervisionados

Nessa seção iremos discutir os resultados da acurácia alcançada por classificadores supervisionados e combinações dos classificadores SVM, *Random Forest*, *Logistic Regression* e *Naïve Bayes*. Tal combinação foi feita selecionando os 3 classificadores de melhor acurácia em cada um dos pré-processamentos utilizados e utilizaremos a votação majoritária dos 3 classificadores para classificar sentimentos nos *tweets*.

Vale ressaltar que, em ambos os pré-processamentos, foi utilizado o subnível IV e no pré-processamento I a técnica de TF-IDF pois tais técnicas alcançaram resultados melhores nos experimentos anteriores. Vale mencionar que nos classificadores SVM e *Random Forest*, foram utilizados os valores de hiperparâmetros que obtiveram os melhores resultados, tais valores foram apresentados nas seções 4.3.1 e 4.3.2. O conjunto de *Stanford* foi treinado com

1.600.000 *tweets* quando utilizada a estratégia de pré-processamento II e, com 3000 *tweets* quando a estratégia de pré-processamento I foi utilizada.

	estratégia de pré-processamento I				
	Acurácia no conjunto *Stanford* (%)	**Acurácia no conjunto OMD (%)**	**Acurácia No conjunto HCR (%)**	**Acurácia no conjunto OPD (%)**	**Média da acurácia nas bases utilizadas (%)**
SVM	81.61	74.88	76.54	78.19	77.81
Random Forest	81.89	76.87	77.29	79.12	78.79
Logistic Regression	79.94	75.95	75.18	79.06	77.53
Naïve Bayes	49.30	71.20	60.15	66.06	61.68
Ensembles	**83.84**	**77.02**	**78.04**	**82.08**	**80.25**

Tabela 4.12: Classificadores supervisionados e combinações entre os mesmos utilizando o pré-processamento I

Observando os dados da Tabela 4.12 podemos constatar que, utilizando a estratégia de pré-processamento I juntamente com técnicas *ensembles*, conseguimos melhorias sobre o nosso melhor resultado individual de 1.95, 0.15, 0.75, 2.96, 1.46 pontos percentuais nas bases de *Stanford*, OMD, HCR, OPD e no resultado médio, respectivamente.

	estratégia de pré-processamento II				
	Acurácia no conjunto *Stanford* (%)	**Acurácia no conjunto OMD (%)**	**Acurácia No conjunto HCR (%)**	**Acurácia no conjunto OPD (%)**	**Média da acurácia nas bases utilizadas (%)**
SVM	**86.90**	70.13	72.93	73.02	75.74
Random Forest	82.72	70.75	73.08	74.08	75.15
Logistic Regression	85.51	63.89	68.87	73.03	72.85
Naïve Bayes	80.22	**77.92**	76.09	78.02	78.06
Ensembles	85.23	76.54	**76.39**	**79.32**	**79.37**

Tabela 4.13: Classificadores supervisionados e combinações entre os mesmos utilizando o pré-processamento II

Analisando a Tabela 4.13 podemos constatar que, utilizando a estratégia de pré-processamento II juntamente com técnicas *ensembles*, conseguimos melhorias sobre os nossos melhores resultados individuais em duas das quatro bases utilizadas (0.30 e 1.30 pontos percentuais nos conjuntos de HCR e OPD respectivamente) e no resultado médio (1.31 pontos percentuais).

Nossos resultados parecem coincidir com outros estudos que utilizaram técnicas *ensembles* em seus trabalhos. Em [101] foi utilizado os classificadores SVM, *Naïve Bayes* e *MaxEnt* (Máxima Entropia) para classificar sentimentos em negativo, neutro e positivo utilizando uma base de dados sobre revisão de filmes. Os classificadores foram utilizados separadamente e combinados. Os autores conseguiram melhorar entre 3 e 4 pontos percentuais a acurácia obtida com classificadores de aprendizado de máquina tradicionais utilizando *ensembles* de classificadores.

Em [102] os autores elaboraram *ensembles* de classificadores utilizando os classificadores individuais *Naïve Bayes*, SVM, *Bayesian Netwo*rk, *C4.5 Decision Tre*e e *Random Forest*. Os classificadores separados e combinados foram utilizados para classificar o sentimento em negativo, neutro e positivo de uma base de dados de *tweets* relacionados a

revisão de serviços aéreos e conseguiam uma melhora na acurácia utilizando a técnica *ensemble* proposta com valores entre 2 e 5 pontos percentuais sobre o melhor resultado obtido entre classificadores individuais.

Existem alguns motivos que explicam o porquê de combinações de classificadores melhorarem o desempenho de classificadores individuais em problemas de classificação. Podemos considerar que alguns deles mencionados em [103] se aplicam ao nosso caso.

O primeiro motivo seria estatístico, suponha que tenhamos vários classificadores diferentes e que todos eles forneçam um bom modelo de classificação baseado no conjunto de treinamento. Se um único classificador for escolhido dentre os disponíveis, pode ocorrer do mesmo não produzir o melhor a generalização necessária para um bom desempenho no conjunto de testes.

O segundo motivo seria computacional, muitos algoritmos de aprendizado funcionam buscando pontos de máximo, tais pontos de máximo podem considerar apenas máximos locais que podem estar longe de ser máximos globais.

Uma terceira razão seria representacional, se um modelo escolhido não puder representar adequadamente a fronteira de decisão para separação de classes, conjuntos de classificadores com modelos diversificados podem representar essas fronteiras de uma forma melhor. Certos problemas são muito difíceis para um determinado classificador resolver. Às vezes, a fronteira de decisão que separa dados de classes diferentes pode ser muito complexa e uma combinação apropriada de classificadores pode tornar possível lidar com esses tipos de problemas.

4.5 Comparativo de Melhores Resultados com Resultados do Estado da Arte

Nessa seção iremos comparar nossos melhores resultados com resultados obtidos pelo estado da arte até o final de 2016 e até final de 2017. Fizemos essa escolha pelo fato de nossos experimentos terem sido feitos tentando melhorar a acurácia obtida pelos modelos que obtiveram os melhores resultados até o final de 2016. Entretanto, no final de 2017 fizemos uma revisão da literatura com o intuito de atualizarmos os dados com os quais estávamos comparando nossos resultados.

Na tabela 4.14 apresentamos as melhores acurácias obtidas no nosso estudo, o pré-processamento e o classificador utilizado para obtenção dessa acurácia. Mostramos, também, a melhor acurácia encontrada na literatura no final de 2016 e de 2017. Tudo isso em cada uma das bases de *Stanford*, OMD, HCR e OPD. Vale mencionar que destacamos, em negrito, o melhor resultado obtido em cada base.

	acurácia obtida no melhor resultado do nosso estudo (%)	**pré-processamento empregado**	**classificador utilizado**	**acurácia obtida pelo estado da arte-final de 2016 (%)**	**acurácia obtida pelo estado da arte-final de 2017 (%)**
Stanford	86.90	pré-processamento II (subnível IV)	SVM	87.20	**87.20**
OMD	**77.02**	pré-processamento I (subnível IV)	*Ensembles*	76.81	76.81
HCR	78.04	pré-processamento I (subnível IV)	*Ensembles*	76.99	**80.90**
OPD	82.08	pré-processamento I (subnível IV)	*Ensembles*	-	**84.00**

Tabela 4.14: Tabela com os melhores resultados obtidos nesse estudo, pré-processamento utilizado, classificador utilizado e acurácias obtidas por modelos que alcançaram o estado da arte no final do ano de 2016 e 2017.

Como podemos constatar na tabela 4.14, conseguimos superar o desempenho em três das quatro bases utilizadas e que foram citados, até o fim de 2016 como estado da arte nesse domínio. Os experimentos mencionados podem ser encontrados em [38], onde os autores conseguiram acurácias de 76.99% na base de HCR e 76.81% na base de OMD; já na base OPD não haviam trabalhos que a utilizaram na época; nosso trabalho conseguiu, nessa base, 82.08% de acurácia. Embora, no conjunto de *Stanford,* não termos conseguido superar o melhor resultado encontrado até então, conseguimos um resultado competitivo ficando apenas 0.30 pontos percentuais atrás do melhor resultado encontrado até então [74].

Em 2017, os autores de [81] superaram nossa acurácia na base HCR, conseguindo uma acurácia de 80.90% contra 78.04% do nosso melhor resultado; e um modelo criado na competição *kaggle* citado em [83] conseguiu um resultado de 84.00% contra o nosso melhor resultado de 82.08%.

Conseguimos, portanto, na data atual apenas ter superado o estado da arte na base de OMD e ficar bem próximo do estado da arte nas outras bases utilizadas.

Capítulo 5
Conclusão e Trabalhos Futuros

Nesse trabalho analisamos a influência de dois tipos de pré-processamento, da técnica TF-IDF, do volume do conjunto de treinamento e de técnicas *ensembles* na acurácia de classificadores supervisionados quando utilizados em problemas de análise de sentimentos, classificando o sentimento de um dado *tweet* em positivo ou negativo.

Os classificadores supervisionados utilizados foram SVM, *Logistic Regression*, *Random Forest* e *Naïve Bayes.* As bases de dados utilizadas foram as bases de *Stanford*, OMD (*Obama-McCain Debate*) e HCR (*Health Care Reform*) e OPD (*First Old Party Debate*).

Concluímos que a adição de pré-processamento contribui de forma substantiva para melhoria na acurácia de classificadores supervisionados. Observamos que diferentes tipos de pré-processamentos contribuem de formas diferentes para os ganhos em acurácia e que o conjunto de atributos que mais contribuiu para essa melhoria no problema desse estudo é o conjunto de *unigrams* e/ou *bigrams* positivos e negativos. Verificamos que o conjunto que menos contribuiu para melhoria na acurácia é o conjunto de *hashtags* e *emoticons*.

Pudemos constatar, também, que diferentes classificadores respondem de diferentes formas aos diferentes tipos de pré-processamentos dependendo da natureza desses classificadores

Testes utilizando técnicas TF-IDF foram feitos e pouco ganho na acurácia foi conseguido utilizando tais técnicas. Tal fato foi atribuído ao fato de muito pré-processamento já ter sido feito e muito já ter sido ganho com tal pré-processamento. Pudemos observar ganhos significativos na acurácia quando aumentamos o volume de dados. No mesmo conjunto de dados (conjunto de *Stanford*) quando aumentamos o numero de *tweets* no treinamento conseguimos uma melhoria significante em pontos percentuais.

Podemos afirmar, que as principais contribuições desse trabalho foram a de avaliar, de forma incremental, diferentes técnicas de pré-processamento, o volume do conjunto de treinamento e de técnicas *ensembles* no desempenho de classificadores supervisionados quando

utilizados para classificar sentimentos em *tweets*. Vale mencionar que, na época em que os experimentos foram realizados (meados de 2016), conseguimos superar o desempenho em três das quatro bases utilizadas e que foram citados, até início de 2017 como estado da arte nesse domínio.

Os experimentos mencionados podem ser encontrados em [38], onde os autores conseguiram acurácias de 76.99% na base de HCR e 76.81% na base de OMD (conseguimos 78.04% na base HCR e 77.02% na base OMD); já na base OPD não haviam trabalhos que a utilizaram na época e nosso trabalho conseguiu, nessa base, 82.08% de acurácia.

Em 2017, os autores de [81] superaram nossa acurácia na base HCR, conseguindo uma acurácia de 80.90% contra 78.04% do nosso melhor resultado; e um modelo criado na competição *kaggle* citado em [83] conseguiu um resultado de 84.00% contra o nosso melhor resultado de 82.08%.

Conseguimos, portanto, na data atual apenas ter superado o estado da arte na base de OMD e ficar bem próximo do estado da arte nas outras bases utilizadas.

Em trabalhos futuros seria interessante explorar a estratégia de *deep learning*. Apesar de abordagens híbridas, que combinam métodos baseados em léxico e aprendizado de máquina, terem obtido um alto desempenho, tais métodos tem o problema de ser necessário definir um conjunto de atributos adequado para cada domínio específico [81].

Os modelos de *deep learning* são diferentes dos métodos tradicionais de aprendizado de máquina na medida em que em um modelo de *deep learning* não é necessário selecionar atributos porque os atributos são extraídos durante o processo do treinamento. O uso dos métodos de *deep learning* tem obtido ótimos resultados no problema de análise de sentimentos [81].

Não exploramos subclasses de sentimento, abordagem que estaria relacionada a suavizar a classificação de sentimentos. Poderíamos dividir as classificações de opinião, por exemplo, em cinco classes (muito negativo, negativo, neutro, positivo, muito positivo) ao invés de duas classes (positivo e negativo) como feito nesse trabalho. Essa área de pesquisa é conhecida como *opinion strength* [111].

Para trabalhos futuros seria interessante, também, explorar alguma abordagem utilizada para classificação de sentimentos que seja dinâmica e capaz de manipular com eficiência fluxos contínuos de *tweets*. Essa abordagem não foi considerada nessa dissertação já que utilizamos conjuntos de dados estáticos. Fluxo de dados em análise de sentimentos vêm sendo muito estudado [112] [113] [114].

Referências Bibliográficas

[1] Liu, Bing, "Sentiment Analysis and Opinion Mining", 1st edtion, Morgan & Claypool Publishers (2012).

[2] Nasukawa, T. and Jeoghee, Y., "Sentiment Analysis: Capturing favorability using natural language processing" in *Proceedings of the KCAP-03, 2nd Intl. Conf. on Knowledge Capture*, Sanibel Island, 2003, pp. 70-77.

[3] Kushal, D. and Lawrence, S., "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews" in *Proceedings of international Conference on World Wide Web,* Princenton, 2003, pp. 519-528.

[4] Das, S. and Chen, M., "Yahoo! for Amazon: Extracting market sentiment from stock message boards" in *Proceedings of APFA-2001*, Bangkok, 2001, pp. 380-393.

[5] Morinaga, S., Yamanishi, K. and Fukushima, t., "Mining products reputation on web" in *Proceedings of ACM SIGKDD International Conference on Knwoledge Discovery and Data Mining (KDD-2002)*, New York, 2002, pp. 341-349.

[6] Bo, P., Lee, L. and Vaithyanathan, S, "Thumbs up? Sentiment classification using machine learning techniques" in *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, San Jose, 2002, pp. 79-86.

[7] Tong, R., "An operational system for detecting and tracking opinions in on-line discussion" in *Proceedings of SIGIR Workshop on Operational Text Classification*, Bangkok, 2001, pp. 35-47.

[8] Turney, P., "Thumbs up or Down? semantic orientation applied to unsupervised classification of reviews" in *Proceedings of Annual Meeting of the Association for Computional Linguistics*, Ottawa, 2002, pp. 417-424.

[9] Vasileios, H. And Wieber, J., "Effects of adjective orientation and gradability on sentence subjective" in *Proceedings of International Conference on Computational Linguistics (COLING-2000)*, New York, 2000, pp. 376-391.

[10] "Sentiment strength detection for the social Web", accessed April 17, 2018, http://sentistrength.wlv.ac.uk/

[11] "MALLET: A Machine Learning for Language Toolkit.", accessed April 18, 2018, http://mallet.cs.umass.edu

[12] Vasileios, H., Klavans, J. and McKeown, M., "Predicting the semantic orientation of adjectivies" in *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-1997)*, New York, 1997, pp. 174-181.

[13] Hearst, M., "Direction-based text interpretation as an information access refinement" in *Text-Based Intelligent Systems*, Lawrence Erlbaum Associates, P. Jacobs Editor, 1992, pp. 257-274.

[14] Wiebe, J., "Identifying subjective characteres in narrative" in *Proceedings of the International Conference Computional Linguistics (COLING-1990)*, Toronto, 1990, pp. 401-406.

[15] Wiebe, J., "Tracking point of view in narrative" in *Computational Linguistics*, New Mexico, 1994, pp. 233-287.

[16] Wiebe, J., Rebecca, B. and O'Hara, T., "Development and use of a gold-standard data set for subjective classifications" in *Proceedings of the Association for Computational Linguistics (ACL-1999)*, Stroudsburg, 1999, pp. 246-253.

[17] Miller, M. et al., "Sentiment Flow Through Hyperlink Networks" in *Proceedingg the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM-2011)*, Stanford, 2011, pp. 312-317.

[18] Mohammad, S. and Tony Y., "Tracking Sentiment in Mail: How Genders Differ on Emotional Axes" in *Proceedings of the ACL Workshop on ACL 2011 Workshop on*

Computational Approaches to Subjectivity and Sentiment Analysis (WASSA-2011), Portland, 2011, pp. 70-79.

[19] Mohammad, S., "From Once Upon a Time to Happily Ever After: Tracking Emotions in Novels and Fairy Tales." in *Proceedings of the ACL 2011 Workshop on Language Technology for Cultural Heritage*, Portland, 2011, pp. 105-114.

[20] Bollen, J., Mao, H., and Zeng, X., "Twitter mood predicts the stock market" in *Journal of Computational Science*, 2011.

[21] Feldman, R., Rosenfeld, B., Bar-Haim and R., Fresko, M., "The Stock Sonar - Sentiment Analysis of Stocks Based on a Hybrid Approach" in *Proceedings of 23rd IAAI Conference on Artificial Intelligence (IAAI-2011)*, San Francisco, 2011, pp. 1642-1647.

[22] Zhang, W., and Skiena, S., "Trading strategies to exploit blog and news sentiment" in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM-2010)*, Washington, 2010, pp 375-378.

[23] Sakunkoo, P. and Sakunkoo, N., "Analysis of Social Influence in Online Book Reviews" in *Proceedings of third International AAAI Conference on Weblogs and Social Media (ICWSM-2009)*, San Jose, 2009, pp. 308-310.

[24] Groh, G. and Hauffa, J., "Characterizing Social Relations Via NLP- based Sentiment Analysis" in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM-2011)*, Barcelona, 2011, pp. 502-505.

[25] Lu, Y., Castellanos, M., Dayal, U., and Zhai, C., "Automatic construction of a context-aware sentiment lexicon: an optimization approach" in *Proceedings of the 20th international conference on World wide web (WWW-2011)*, Hyderabad, 2011, pp. 347-356.

[26] Archak, N., Ghose, A., and Ipeirotis, P., "Show me the money!: deriving the pricing power of product features by mining consumer reviews" in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD-2007)*, New York, 2007, pp. 56-65.

[27] Chen, Y. and Xie, J., "Online consumer review: Word-of-mouth as a new element of marketing communication mix" in *Management Science*, 2008, pp. 477-491.

[28] Das, S, and Chen M., "Yahoo! for Amazon: Extracting market sentiment from stock message boards" in *Proceedings of APFA-2001*, Barcelona, 2001, pp. 1375-1388.

[29] Dellarocas, C., Zhang X., and Awad, N., "Exploring the value of online product reviews in forecasting sales: The case of motion pictures" in *Journal of Interactive Marketing*, 2007, pp. 23-45.

[30] Hu, N., Pavlou, P., and Zhang, J., "Can online reviews reveal a product's true quality?: empirical findings and analytical modeling of Online word-of-mouth communication" in *Proceedings of Electronic Commerce (EC-2006)*, New York, 2006, pp. 324-330.

[31] Park, D., Lee, J., and Han, I., "The effect of on-line consumer reviews on consumer purchasing intention: The moderating role of involvement" in *International Journal of Electronic Commerce*, 2007, pp. 125-148.

[32] Giachanou, A and Crestani, F., "Like It or Not: A Survey of Twitter Sentiment Analysis Methods" in *ACM Computing Surveys Journal*, Volume 49 Issue 2, Article nº., 2016.
59-60-62-123-124

[33] Mohammad, S., Kiritchenko, S. and Zhu, X., "NRC-Canada: Building the state-of- the-art in sentiment analysis of tweets" in *Proceedings of the 7th International Workshop on Semantic Evaluation - 2nd Joint Conference on Lexical and Computational Semantics (SemEval'13)*, Atlanta, 2013, pp. 321-327.

[34] Nakov, P., Kozareva, Z., Ritter, A., Rosenthal, S., Stoyanov, V., and Wilson, T., "Semeval-2013 task 2: Sentiment analysis in twitter" in *Proceedings of the 7th International Workshop on Semantic Evaluation - 2nd Joint Conference on Lexical and Computational Semantics (SemEval'13)*, Atlanta, 2013, pp. 312-320.

[35] Pagolu, V., Challa, K. and Panda, G., "Sentiment Analysis of Twitter Data for Predicting Stock Market Movements" in *International conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, 2016, Paralakhemundi, pp. 1345-1350.

[36] Alayba, A., Palade, V., England, M. and Iqbal, R., "Improving Sentiment Analysis in Arabic Using Word Representation" in *Proc. 2nd International Workshop on Arabic Script Analysis and Recognition,* 2018, London, pp. 4134-4138.

[37] Hamdan, H., Bechet, F., and Bellot, P., "Experiments with DBpedia, WordNet and SentiWordNet as resources for sentiment analysis in micro-blogging" in *Proceedings of the 7th International Workshop on Semantic Evaluation - 2nd Joint Conference on Lexical and Computational Semantics (SemEval'13), Vol. 2. Association for Computational Linguistics*, Atlanta, 2013, pp. 455-459.

[38] da Silva, N., Hruschka, E. and Hruschka Jr, E., "Tweet sentiment analysis with classifier ensembles" in *Decision Support Systems 66 (2014)*: 170-179.

[39] Yan, Y., Yang, H. and Wang, H., "Two Simple and Effective Ensemble Classifiers for Twitter Sentiment Analysis" in 2017 Computing Conference, 2017, London, pp. 1386-1393.

[40] Aston, N., Liddle, J., and Hu, W., "Twitter sentiment in data streams with perceptron" in *Journal of Computer and Communications* 3 (2014): pp. 11-16.

[41] Maas, A., Daly, E., Pham, P., Huang, D., Andrew Y. and Potts, C., "Learning word vectors for sentiment analysis" in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT'11). Association for Computational Linguistics*, 2011, Stroudsburg, pp. 142-150.

[42] Tang, D., Qin, B., and Liu, T., "Learning semantic representations of users and products for document level sentiment classification" in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (ACL'15). The Association for Computer Linguistics*, Beijing, 2015, pp. 1014-1023.

[43] Tang, D., Qin, B., Liu, T., and Yang, Y., "User modeling with neural network for review rating prediction" in *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15). AAAI Press*, Buenos Aires, 2015, pp. 1340-1346.

[44] Dey, K., Shrivastava, R., Kaushik, S., "Topical Stance Detection for Twitter: A Two-Phase LSTM Model Using Attention" in *40th European Conference on Information Retrieval (ECIR)*, 2018, Grenoble.

[45] Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., and Xu, K., "Adaptive recursive neural net- work for target-dependent twitter sentiment classification" in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2014, Stroudsburg, pp. 49-54.

[46] Becker, W., Wehrmann, J., Cagnini and H., Barros, R., "An Efficient Deep Neural Architecture for Multilingual Sentiment Analysis in Twitter" in *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference*, 2017, Marco Island, pp. 246-251.

[47] D, Duy-Tin. and Zhang, Y., "Target-dependent twitter sentiment classification with rich automatic features" in *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 2015, Buenos Aires, pp. 1347-1353.

[48] Thelwall, M., Buckley, K., Paltoglou, G., and Kappas, A., "Sentiment strength detection in short informal text" in *Journal of the American Society for Information Science and Technology* 61 (2010): pp. 2544-2558.

[49] Thelwall, M., Buckley, K. Paltoglou, G. "Sentiment strength detection for the social web" in *Journal of the American Society for Information Science and Technology* 63 (2012): pp. 163-173.

[50] Reckman, H., Baird, C., Crawford, J., Crowell, R., Micciulla, L., Sethi, S. and Veress, F., "Rule-based detection of sentiment phrases using SAS sentiment analysis" in *2nd Joint Conference on Lexical and Computational Semantics* 2 (2013): pp. 513-519.

[51] Go, A., Bhayani, P. and Huang, L., "Twitter Sentiment Classification Using Distant Supervision" in *Technical Report*, Standford, 2009.

[52] Shamma, D., Kennedy, L. and Churchill, E., "Tweet the debates: Understanding community annotation of uncollected sources" in *Proceedings of the First SIGMM Workshop on Social Media (WSM'09)*, 2009, New York, pp. 3-10.

[53] Saif, H., He, Y., Fernandez, M. and Alani, H., "Contextual semantics for sentiment analysis of twitter" in *Information Processing and Management: an International Journal* 52 (2016): pp. 5-19.

[54] Speriosu, M., Sudan, N., Upadhyay, S. and Baldridge, J., "Twitter polarity classification with label propagation over lexical links and the follower graph" in *Proceedings of the First Workshop on Unsupervised Learning in NLP (EMNLP'11)*, 2011, Stroudsburg, pp. 53-63.

[55] Saif, H., Fernandez, M., He, Y. and Alani, H, "Evaluation datasets for twitter sentiment analysis a survey and a new dataset, the STS-gold" in *Proceedings of the 1st International Workshop on Emotion and Sentiment in Social and Expressive Media: Approaches and Perspectives from AI (ESSEM'13)*, 2013, Turin, pp. 91-100.

[56] Dey, A., Jenamani, J. and Thakkar, J, "Senti-N-Gram: An n-gram lexicon for sentiment analysis" in *Elsevier Journal*, 2018, pp. 92-105.

[57] Hu, X., Tang, J., Gao, H. and Liu, H., "Unsupervised sentiment analysis with emotional signals" in *Proceedings of the 22nd International Conference on World Wide Web (WWW'13)*, New York, 2013, pp. 607-618.

[58] Jianqiang, Z. and Xiaolin, G., "Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis" in *IEEE Access*, Volume 5, 2017, pp. 2870-2879.

[59] Ghiassi, M., Skinner, J. and Zimbra, D., "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network" in *Expert Systems with Applications: An International Journal* 40 (2012): pp. 6266-6282.

[60] Ismail, E., Harous, S. and Belkhouche, B., "A Comparative Analysis of Machine Learning Classifiers for Twitter Sentiment Analysis" in *Conference: 17th International Conference on Intelligent Text Processing and Computational Linguistics,* Konya, 2016, pp. 226-231.

[61] Khan, F., Bashir, S. and Qamar, U., "Twitter opinion mining framework using hybrid classification scheme" in *Decision Support Systems Journal* 57 (2014): pp. 245-257.

[62] Speriosu, M., Sudan, N., Upadhyay, S. and Baldridge, J., "Twitter polarity classification with label propagation over lexical links and the follower graph" in *Proceedings of the First Workshop on Unsupervised Learning in NLP (EMNLP'11)*, 2011, Stroudsburg, pp. 53-63.

[63] Cui, A., Zhang, M., Liu, Y. and Ma, S., "Emotion tokens: Bridging the gap among multilingual twitter sentiment analysis" in *Proceedings of the 7th Asia Conference on Information Retrieval Technology (AIRS'11)*, 2011, Berlin, pp. 238–249.

[64] Wang, X., Wei, F., Liu, X., Zhou, M. and Zhang, M., "Topic sentiment analysis in twitter: A graph-based hashtag sentiment classification approach" in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11)*, 2011, New York, pp. 1031–1040.

[65] Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M. and Li, P., "User-level senti- ment analysis incorporating social network" in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11)*, 2011, New York, pp. 1397-1405.

[66] Kontopoulos, E., Berberidis, C., Dergiades, T. and Bassiliades, N., "Ontology- based sentiment analysis of twitter posts" in *Expert Systems with Applications: An International Journal* 40 (2013): pp. 4065-4074.

[67] Korenek, P. and Simko, M., "Sentiment analysis on microblog utilizing appraisal theory" in *World Wide Web Journal* 17 (2014): pp. 847-867.

[68] Hu, X., Tang, Tang, J. and Liu, "Exploiting social relations for sentiment analysis in microblogging" in *Proceedings of the 6th ACM International Conference on Web Search and Data Mining (WSDM'13)*, 2013, New York, pp. 537-546.

[69] "Data for everyone", accessed July 17, 2017, https://www.crowdflower.com/data-for-everyone/

[70] Sun, F. and Belatreche, D., "Pre-processing Online Financial Text for Sentiment Classification: A Natural Language Processing Approach" in *Computational Intelligence for Financial Engineering & Economics Conference,* 2014, London, pp. 446-460.

[71] Zhang, L. et al, "Combining lexicon-based and learning based methods for twitter sentiment analysis" in *International Journal of Electronics, Communication and Soft Computing Science & Engineering* (2011): pp. 89.

[72] Agarwal, A., Xie, B., Vovsha, L., Rambow, O., and Passonneau, R., "Sentiment analysis of twitter data" in *Proceedings of the Workshop on Languages" in Social Media (LSM'11). Association for Computational Linguistics*, Stroudsburg, 2011, pp. 30-38.

[73] Kazi, A. et al, "Adding Emotional Tag to Augment Context-Awareness in Social Network Services" in *IEEE International Instrumentation and Measurement Technology Conference*, 2011, Makkah Al Mukahhamah, pp. 15-21.

[74] Bakliwal, A. et al, "Mining Sentiments from Tweets" in *Proceedings of the 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, Jeju, 2013, pp. 11-18.

[75] "Amazon EC2 pricing", accessed July 17, 2017, https://aws.amazon.com/ec2/pricing/on-demand/

[76] "Bluemix Blog", accessed July 17, 2017, https://www.ibm.com/blogs/bluemix/2017/05/try-dashdb-transactions-7-days-free-month/

[77] "RStudio Server Amazon Machine Image (AMI)", accessed July 17, 2017, http://www.louisaslett.com/RStudio_AMI/

[78] "Package 'e1071'", accessed July 17, 2017, https://cran.rproject.org/web/packages/e1071/e1071.pdf

[79] "Package 'LOGIT'", accessed July 17, 2017, https://cran.r-project.org/web/packages/LOGIT/LOGIT.pdf

[80] "Class Material #05", accessed April 15, 2018, http://www.ic.unicamp.br/~rocha/teaching/2015s2/mo444/classes/mo444-class-materials-05.pdf

[81] Nguyen, H. and Nguyen, M., "A Deep Neural Architecture for Sentence-level Sentiment Classification in Twitter Social Networking" in *Conference of the Pacific Association for Computational Linguistics*, 2017, Yangon, pp. 15-17.

[82] Kauer, A. and Moreira, V., "Using Information Retrieval for Sentiment Polarity Prediction" in *International Expert Systems with Applications: An International Journal,* Volume 61, 2016, New York, pp. 282-289.

[83] Ganguli, S., Dunnmon, J. and Husic, B., "Predicting State-Level Agricultural Sentiment with Tweets from Farming Communities" in *Stanford CS 221 Artificial Intelligence Project Report*, 2017.

[84] "LSTM Sentiment Analysis | Keras ", accessed September 8, 2017, https://www.kaggle.com/ngyptr/lstm-sentiment-analysis-keras

[85] Kouloumpis, E., Wilson, T. and Moore, J., "Twitter sentiment analysis: The good the bad and the omg!" in *Proc. the Fifth International AAAI Conference on Weblogs and Social Media*, 2011, Barcelona, pp.538-541.

[86] Terrana, D., Augello, A. and Pilato, G., "Automatic Unsupervised Polarity Detection on a Twitter Data Stream" in *Proc. 2014 IEEE International Conference on Semantic Computing*, 2014, Newport Beach, pp.128-134.

[87] Saif, H., He, Y., Fernandez, M. and Alani, H., "Semantic Patterns for Sentiment Analysis of Twitter" in *Proc. the 13th International Semantic Web Conference*, Springer International Publishing, 2014, Trentino, pp.324-340.

[88] Feldman, R., "Techniques and applications for sentiment analysis" *in Communications of the ACM,* Volume 56, Issue 4, 2013, pp. 82-89.

[89] Saif, H., Fernandez, M., He, Y. and Alani, H., "On Stopwords, Filtering and Data Sparsity for Sentiment Analysis of Twitter" In *Proc. 9th Language Resources and Evaluation Conference (LREC)*, 2014, Reykjavik, pp.80-81.

[90] Zheng, L., Wang, H. and Gao, S., "Sentimental feature selection for sentiment analysis of Chinese online reviews" in *International Journal of Machine Learning and Cybernetics*, Volume 9, Issue 1, 2018, pp. 75-84.

[91] Balahura, A. and Turchib M., "Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis" in *Computer Speech & Language*, Volume 28, Issue 1, 2014, pp. 56-75.

[92] Li, X., Wang, H., Bin, U. and Ling, C., "Data Sparseness in Linear SVM" in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 2015, Buenos Aires, pp. 3628-3634.

[93] Kavalec M. and Strossa, P., "Text Classification by Bootstrapping with Keywords, EM and Shrinkage" *in Unsupervised Learning in Natural Language Processing*, 2002, pages 52-58.

[94] Huang, L., Jin, Y. and Shen, D., "Longitudinal clinical score prediction in Alzheimer's disease with soft-split sparse regression based random forest" in *Neurobiology of aging*, Volume 46, pp. 180-191.

[95] Takwoingi, Y., Guo, B., Riley, R. and Deeks, J., "Performance of methods for meta-analysis of diagnostic test accuracy with few studies or sparse data" in *Statistical Methods in Medical Research*, Volume 26, 2017, pp. 1897-1911.

[96] Pak, A. and Paroubek, P., "Twitter as a corpus for sentiment analysis and opinion mining" in *Proc. LREC*, Volume 10, 2010, pp.1320-1326.

[97] Badr, B. and Fatima, S., "Using Skipgrams, Bigrams, and Part of Speech Features for Sentiment Classification of Twitter Messages" in *proceedings of the 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, 2015, Jeju, pp. 11-18

[98] Baeza-Yates, R. and Liaghat, Z., "Quality-Efficiency Trade-offs in Machine Learning for Text Processing" in *2017 IEEE International Conference on Big Data*, 2017, Boston, pp. 897-904.

[99] Vo, H., Lam, H., Nguyen, D. and Tuong, N., "ASTD: Arabic Sentiment Tweets Dataset" in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, Lisbon, pp. 2515-2519.

[100] Nail, M., Aly, M. and Amir, A., "Topic Classification and Sentiment Analysis for Vietnamese Survey System" in *Asian Journal of Computer Science and Information Technology*, Volume 6, 2016, pp. 27-34.

[101] Kanakaraj, M. and Guddeti, R., "Performance Analysis of Ensemble Methods on Twitter Sentiment Analysis using NLP Techniques" in *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, Anaheim, 2015, pp. 169-170.

[102] Wan, Y. and Gao, Q., "An Ensemble Sentiment Classification System of Twitter Data for Airline Services Analysis" in *2015 IEEE 15th International Conference on Data Mining Workshops,* 2015, Atlantic City, pp. 1318-1325.

[103] Dietterich, T. "Ensemble methods in machine learning" in *Proceedings of the First International Workshop on Multiple Classifier Systems*, 2000, London, pp. 1-15.

[104] Jindal, N. and Liu, B., "Review spam detection" in *Proceedings of WWW (Poster paper)*, 2007.

[105] McGlohon, M., Glance, N., and Reiter, Z., "Star quality: Aggregating reviews to rank products and merchants" in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM- 2010)*, Washington, 2010, pp. 114-121.

[106] O'Connor, B., Balasubramanyan, R., Routledge, R. and Smith, N., "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. Series.Smith. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series." in *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM 2010)*, Washington, 2010, pp. 122-129.

[107] Andranik, T., Sprenger, T., Sandner, P. and Welpe, I., "Predicting elections with twitter: What 140 characters reveal about political sentiment" in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM-2010)*, Pennsylvania, 2010, pp. 178-185.

[108] Ni, X., Sun, J., Chen, Z., Yang, Q., "Cross-domain sentiment classification via spectral feature alignment" in *Proceedings of International Conference on World Wide Web (WWW-2010)*, New York, 2010, pp. 751-760.

[109] Yano, T. and Noah S., "What's Worthy of Comment? Content and Comment Volume in Political Blogs" in *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM 2010)*, Washington, 2010, pp. 359–362.

[110] Asur, S. and Huberman, B., "Predicting the future with social media." in *Arxiv preprint arXiv:1003.5699*, 2010.

[111] Turney, D. and Littman, M., "Measuring praise and criticism: Inference of semantic orientation from association" in *ACM Trans. Inf. Syst.*, 2003 v.21, n.4, pp. 315-346.

[112] Lourenco Jr., R., Veloso, A., Pereira, A., Meira Jr., W., Ferreira, R. and Parthasarathy, S., "Economically-efficient sentiment stream analysis" in *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR '14*, 2014, New York, pp. 637-646.

[113] Kim, G., Lee, S. and Kyeong, S., "Discovering hot topics using twitter streaming data: Social topic detection and geographic clustering" in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, *ASONAM '13*, 2013, New York, pp. 1215-1220.

[114] Mejova, Y. and Srinivasan, P., "Political speech in social media streams: Youtube comments and twitter posts" in *Proceedings of the 4th Annual ACM Web Science Conference, WebSci '12*, 2012, New York, pp. 205-208.

[115] Taboada, M., Brooke, J., Tofiloski, M., Voll, M. and Stede, M., "Lexicon-based methods for sentiment analysis" in *Journal in Computationoal Linguistics* 37 (2011): pp. 267-307.

[116] Ding, X., Liu, B. and Yu, P., "A holistic lexicon-based approach to opinion mining" in *Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM'08)*, 2008, New York, pp. 231-240.

[117] "Computational Linguistics & Psycholinguistics Research Center", accessed April 18, 2018, http://www.clips.ua.ac.be/pattern