Capítulo

1

Análise de Sentimentos Utilizando Técnicas de Classificação Multiclasse

Alexandre de Castro Lunardi, José Viterbo, Flávia Cristina Bernardini

Abstract

After the advent of Web 2.0, finding reviews on products, businesses, services, organizations and many other areas on the web became really easy. These opinions can be found on social networks, blogs and specialized e-commerce sites that provide tools so that a user can evaluate an item. These reviews can be useful in recommender systems, stating whether a product is suitable or not via the existing web assessments. This type of analysis is known as binary classification. Beyond binary classification, other forms may include scales evaluation, problem known as multiclass classification. An example of this type of classification is ratings inference (multiclass classification). This work intends to introduce the feelings analysis process, using feature extraction techniques by performing the steps of textual preprocessing, feature selection, vectorization, and finally, machine learning, in order to infer whether an opinion is positive or negative (binary classification) or to infer a rating.

Resumo

Com o advento da Web 2.0, encontrar opiniões sobre produtos, negócios, serviços, organizações e sobre tantos outros domínios é algo comum na web. Essas opiniões podem ser encontradas em redes sociais, blogs especializados e em sites e-commerce que disponibilizam ferramentas para que um usuário possa avaliar um item. Essas opiniões podem ser de grande utilidade em sistemas de recomendação, informando se um produto é recomendado ou não por meio das avaliações existentes na web. Este tipo de análise é baseada em técnicas de classificação binária. Além da classificação binária, outras formas podem incluir a avaliação de escalas, problema conhecido como classificação multiclasse. Um exemplo deste tipo de classificação é a inferência de ratings (classificação multiclasse). Esse estudo tem o intuito de introduzir o processo de análise de sentimentos utilizando técnicas de extração de características, executando as fases de pré-processamento textual, seleção de características,

vetorização e, por fim, o aprendizado de máquina, a fim de inferir se uma opinião é positiva ou negativa (classificação binária) ou inferir um rating.

1.1 Introdução

Opiniões sempre foram úteis no que diz respeito à tomada de decisões dos seres humanos [CAMBRIA et al. 2013]. Nossas escolhas sempre foram, em certo grau, dependentes das opiniões e conselhos de outras pessoas [LIU 2012]. Além disso, é de grande importância para empresas conhecer o sentimento das pessoas em relação a um produto ou serviço, o que permite realizar previsões de mercado ou oferecer recomendações aos consumidores [TURNEY 2002], tornando essas empresas mais próximas de seu público-alvo.

Com o advento da Web 2.0, é cada vez mais fácil encontrar opiniões valiosas relacionadas a produtos, serviços, organizações, indivíduos, eventos e vários outros domínios. Isso se deve ao crescente uso de redes sociais, blogs e, principalmente, ferramentas que permitem aos usuários deixar registrado seus comentários sobre algum produto ou serviço em sites de comércio eletrônico. Essa crescente disponibilização de dados é também conhecida como "web social" [CAMBRIA *et al*. 2013]. Isso pode ser notado em sites de comércio eletrônico como o *Booking.com*™[1] e a *Amazon*™[2], nos quais os clientes podem deixar seus comentários, revelando suas opiniões a respeito do produto ou serviço oferecido.

Com essa grande quantidade de informação disponível na Internet, analisar todo o conjunto de opiniões encontradas se tornou uma tarefa inviável para o ser humano. Com isso, capturar e processar de forma adequada essas informações por meio de técnicas computacionais – a chamada mineração de opiniões ou análise de sentimentos [CAMBRIA *et al*. 2013] – se torna fundamental para permitir a identificação do real interesse do público sobre algum item. A comunidade científica vem, dessa forma, desenvolvendo ferramentas que visam auxiliar na recuperação e tratamento de opiniões – ou avaliações – sobre produtos e serviços, disponíveis na web social. Dessa forma, pesquisas sobre mineração de opiniões e/ou análise de sentimentos são uma das áreas mais ativas e desafiantes, abordadas principalmente na área de Processamento de Linguagem Natural (NLP).

Análise de sentimentos ou *mineração de opiniões* são os principais termos empregados para descrever a análise automática de textos subjetivos, isto é, textos que contém não apenas fatos ou explicações técnicas sobre algo, mas alguma opinião a respeito de um item. A partir dessa análise, é possível identificar aspectos distintos de um item, por exemplo, a localização ou a limpeza de um hotel, a durabilidade ou a facilidade de uso de um eletrodoméstico. É possível realizar também uma análise agregada das avaliações sobre um determinado item, identificando o sentimento geral em relação a esse item. Com isso, poderemos saber, por exemplo, se um hotel é recomendado pelos consumidores que ali se hospedaram, de acordo com o conjunto de opiniões emitidas. Ou seja, considerando-se apenas uma escala binária, os aspectos específicos de um produto ou serviço podem ser classificados como bons ou ruins. Por exemplo, pode-se avaliar se a câmera de um celular é recomendável ou não, ou se a localização de um hotel é boa ou

[1] http://www.booking.com/
[2] http://www.amazon.com/

não. A partir da análise de diversos aspectos, chega-se a uma conclusão sobre o sentimento final sobre um item. Por exemplo, após avaliar atributos de um celular como a câmera, facilidade de uso, preço e vários outros aspectos, pode-se chegar a uma conclusão final sobre a recomendação do celular.

Considerando um texto subjetivo que representa a avaliação de um usuário sobre um determinado item, este pode ser classificação utilizando-se técnicas binárias ou multiclasse. Na classificação binária, o objetivo é rotular essa avaliação como positiva ou negativa (boa ou ruim), com relação ao sentimento expressado pelo usuário. Um exemplo de ferramenta criada para analisar o sentimento de uma opinião é a *sentiment140*[3], proposta por [GO; BHAYANI; HUANG 2009]. Nesse site é possível verificar o sentimento em relação a uma entidade (empresa, produto, serviço etc) utilizando *tweets* sobre a entidade em análise. Essa ferramenta seleciona os *tweets* de acordo com a palavra-chave informada pelo usuário e classifica os *tweets* encontrados como positivos ou negativos. Além disso, é apresentado um gráfico com a porcentagem total de tweets positivos e negativos. Outros sites também exemplificam o uso da análise de sentimentos, como o *NLTK Text Classification*[4], no qual o usuário digita um texto sobre algo e o sistema determina se é uma opinião (texto subjetivo) e, caso positivo, se a polaridade desta é positiva ou negativa. Outro exemplo é o site *Skyttle*[5], no qual a opinião informada também é classificada como boa ou ruim e, além disso, as frases com sentimento bom são marcadas em verde e as frases com sentimento ruim são marcadas em vermelho.

A classificação ou análise multiclasse analisa uma avaliação considerando escalas com mais de dois "valores de sentimento", como por exemplo, "bom", "neutro" ou "ruim". Em diversos cenários, os produtos ou serviços são classificados em escalas de valores múltiplos. Pode-se citar como exemplo o site *Booking*, no qual os hotéis são classificados com notas que variam de 0 a 10. Assim sendo, uma forma típica de análise multiclasse é o Problema de Inferência de *Rating* (*Rating*-inference Problem- RIP), baseada em escalas de *rating* que tipicamente variam de 1 a 5 estrelas [PAN e LEE 2005]. Em alguns casos, essa escala pode ser analisada como 4 classes, na qual a classe 3 (neutra) é desconsiderada, ou como 3 classes, nas quais as classes 1 e 2 são unidas, assim como as classes 4 e 5.

Escalas baseadas em *ratings* estão presentes em larga escala em ferramentas de avaliação disponíveis em serviços como *Amazon*™ e *Netflix*™[6], e projetos como o *GroupLens*™[7] [KONSTAN *et al.* 1997], com avaliações utilizando opiniões rotuladas em uma escala 5-*ratings*. A importância da avaliação correta pode ser comprovada de acordo com a pesquisa do site *ComScore*[8], que mostra que os consumidores têm maior disposição para gastar entre 20% e 99% a mais em serviços que tenham uma classificação excelente (5 estrelas) do que um serviço classificado com 4 estrelas (Bom). Para o domínio de hotéis, esse percentual é de 38%. Técnicas de classificação binária não permitiriam que essa divisão (4 e 5 estrelas) fosse identificada automaticamente em trabalhos em análise de sentimentos. Além disso, considerando a grande utilização de várias classes na análise de textos subjetivos, é de grande interesse analisar não somente a polaridade de um item,

[3] http://www.sentiment140.com
[4] http://text-processing.com/demo/sentiment/
[5] http://www.skyttle.com/demoin
[6] http://www.netflix.com
[7] http://grouplens.org
[8] http://comscore.com

mas também avaliar os graus de positividade e negatividade por meio de *ratings* numéricos, baseados, por exemplo, na escala de Likert [LIKERT 1932], variando de 1 a 5 estrelas.

Embora seja uma forma de classificação essencial devido ao grande uso de *ratings* e de grande importância para a comunidade e para empresas, o número de trabalhos disponíveis em análise de sentimentos multiclasse é muito inferior se comparado aos trabalhos com classificação binária. Mesmo em cenários tipicamente de escalas múltiplas, grande parte das análises de opiniões consideram apenas duas classes principais – agrupadas em recomendado ou não recomendado –, em que, em uma escala de 1 a 5 estrelas, por exemplo, 4 ou 5 estrelas são consideradas recomendáveis e 1, 2 ou 3 estrelas não são recomendáveis.

Um possível emprego para as técnicas de análise multiclasse, seria permitir a classificação automática de comentários de usuários em sites de produtos ou serviços, diminuindo o chamado efeito manada (*herding effect*) [WANG e WANG 2014]. Esse efeito acontece na avaliação direta realizada pelos usuários quando estes se deixam influenciar pela avaliação da maioria. Por exemplo, um usuário pode ter achado um celular bom (4 estrelas), mas caso a maioria dos outros usuários tenham considerado excelente (5 estrelas), existe a possibilidade de que ele avalie o celular com base na média dos outros usuários e não no que o celular representou para ele. Além disso, a classificação automática de comentários baseada em análise multiclasse poderia simplesmente evitar os erros da classificação realizada pelo usuário, e casos em que um comentário não condiz o número de estrelas atribuídos, poderiam ser evitados. Mesmo com a divisão com *ratings* 4 e 5 consideradas recomendadas, de acordo com a pesquisa feita pelo site *PracticalECommerce*[9] a inferência de *ratings* seria de grande utilidade tendo em vista que uma estrela a mais ou a menos pode fazer a diferença no momento da compra de um item.

Dessa forma, esse capítulo tem como principal objetivo a apresentação de técnicas de análise de sentimentos baseada em classificação multiclasse. Primeiramente apresentamos os principais conceitos relacionados ao tema. Em seguida, apresentamos as técnicas de extração de características que possibilitam uma boa representação de opiniões na forma de vetores de características. Em seguida, explicamos os diversos modelos e algoritmos de classificação que podem ser utilizados e discutimos métricas para avaliar de forma adequada o desempenho desses algoritmos. Além disso, apresentamos também uma discussão sobre os trabalhos que utilizam aprendizado de máquina na área de análise de sentimentos.

1.2 Definições

Segundo [LIU 2012], a *análise de sentimentos* ou *mineração de opiniões* é o campo de estudo que analisa as atitudes, emoções, sentimentos e as opiniões das pessoas em relação a entidades - como produtos, serviços, organizações, eventos, tópicos - e os atributos dessas entidades. Ela é um campo desafiador na área de Processamento de Linguagem Natural já que trata várias questões de PLN como a tratamento de negação e retirada de palavras-chave e pode cobrir muitos problemas, desde a classificação em relação à polaridade de uma opinião até o processo de sumarização do sentimento geral sobre algo.

[9] http://www.practicalecommerce.com/articles/93017-Study-5-Star-Reviews-Not-Necessarily-Helpful

Seja um texto *d*, a tarefa inicial na mineração de opiniões consiste em determinar se *d* é subjetivo, ou seja, expressa um sentimento. Seja tal texto considerado subjetivo, formalmente, uma opinião pode ser representada como uma *5-tupla* [LIU 2012] O = (e, a, s, h, t), na qual:

- *e* é o nome da entidade ou objeto ao qual uma opinião se refere;
- *a* é o atributo específico da entidade;
- *s* é o sentimento do autor da opinião em relação a um atributo ou entidade;
- *h* é o autor da opinião, e;
- *t* é a data na qual a opinião foi criada.

Como exemplo, temos a seguinte opinião sobre o hotel *H* retirada do site TripAdvisor[1]:

User: DesDeeMona (h)
Title: A magnicent building of fading grandeur, redolent of earlier times
Rating: 4
Date: April 28, 2015 (t)
Review: "Ground floor lobbies and suites with art deco design are impressive. Rooms (a) are spacious (s) with high ceilings and plenty of room in the en suite shower. Yes, lifts are a little slow, the paint is peeling, the plaster cracking, there are stains on the carpet - but hey, everything works, the sheets are well laundered and beds are comfortable".

Nesse exemplo, pode-se observar que o hotel é a entidade e um dos atributos são os quartos, destacados com a letra *(a)* no documento. Eles são classificados pelo autor como "espaçosos", como demarcado acima pela letra *(s)*. Para [DAVE et al. 2003], a tarefa ideal na análise de sentimentos deveria processar um conjunto de opiniões sobre certa entidade, gerando uma lista de atributos para a mesma e agregar opiniões sobre os atributos da entidade. Entretanto, outros autores consideram apenas a entidade da opinião e o sentimento final, como feito em [PANG; LEE; VAITHYANATHAN 2002].

De forma resumida, a taxonomia de análise de sentimentos está presente na Figura 1. A ideia básica para o problema de análise de sentimentos é que um usuário emita uma opinião, também chamada de avaliação ou revisão. Essa avaliação pode ser sobre uma entidade ou item (como um hotel, por exemplo) ou pode ser relativa a um aspecto ou atributo específico de um item (a localização do hotel). Esse sentimento geralmente pode ser classificado em duas ou mais classes. A forma mais comum de classificação é a classificação binária, que diz se uma opinião é positiva ou negativa. Além disso, outras formas de classificação merecem destaque como a classificação por meio de *ratings* (presentes no site da *Amazon*) ou por meio de notas (*Booking.com*).

1.3 A Análise de Sentimentos e o Aprendizado de Máquina

Um dos primeiros trabalhos a analisar o sentimento das pessoas através de dados da web foi discutido em [DAS e CHEN 2001], e utilizou o termo *extração de sentimento* para capturar a influência da opinião de indivíduos no domínio de finanças. Já Pang *et al.* [PANG *et al*. 2002], utilizam o termo *classificação de sentimentos* para avaliar documentos considerando o sentimento geral de uma opinião, classificando-as como positivas ou negativas. Outro trabalho inicial é o de Turney [TURNEY 2002] que visa classificar opiniões como *recomendadas* ou *não recomendadas* (em inglês, *thumbs up* e *thumbs down*). Apenas em Nasukawa e Yi [NASUKAWA e YI 2003) o termo *análise de*

sentimentos é empregado, e assim como em [PANG; LEE; VAITHYANATHAN 2002], os autores introduzem uma pesquisa para classificar uma opinião como positiva ou negativa.

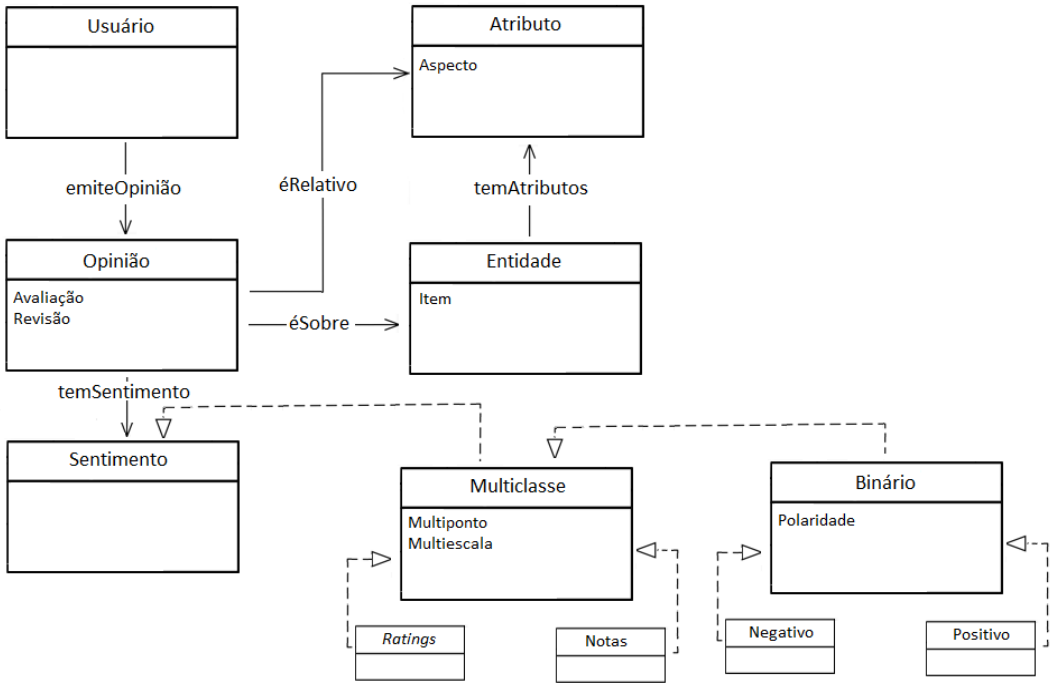

Figura 1. Ontologia com a análise de sentimentos

De acordo com Cambria *et al.* [CAMBRIA *et al*. 2013], a mineração de opiniões pode ser agrupada em quatro campos, na qual a análise pode ser realizada por meio de:

- Palavras-chave e afinidade léxica: classifica o texto de acordo com a presença de palavras sem sentido ambíguo, tais como "feliz", "triste" e "medo". Além de detectar palavras óbvias, também atribui a outras palavras uma relação de afinidade com um sentimento, seja ele bom ou ruim. Um exemplo de aplicação é o *SentiWordNet*[10] 3.0 [BACCIANELLA; ESULI; SEBASTIANI 2010], um recurso léxico criado a fim de orientar aplicações em mineração de opiniões.
- Aprendizado de máquina: utiliza modelos de aprendizado de máquina, como Naive Bayes e Support Vector Machine (SVM), para classificar um texto. Nesse caso, o sistema, além de aprender a importância de uma palavra-chave óbvia, considera outras palavras que podem ser fundamentais, além da possibilidade de analisar a frequência ou a pontuação de um texto.
- Orientação semântica: esses métodos calculam a orientação semântica (por exemplo, para o problema binário saber a polaridade da palavra) de uma palavra baseada na coocorrência da mesma com palavras que possuem a mesma orientação. O principal trabalho que propõe um método que calcule essa orientação semântica é o algoritmo proposto por Turney, 2002 [TURNEY 2002]. O algoritmo Pointwise Mutual Information and Information Retrieval (PMI-IR) é utilizado a fim de medir a similaridade de pares de palavras ou frases. A orientação é calculada pela comparação da similaridade de uma palavra em relação aos sentimentos positivo e negativo.

[10] http://www.sentiwordnet.isti.cnr.it

- Baseado em conceitos: usam ontologias ou redes de palavras-chave para realizar a análise textual. Podem analisar expressões que não possuem uma emoção explícita, mas estão relacionadas a um sentimento implicitamente. No trabalho realizado por Kontopoulos et al [KONTOPOULOS *et al*. 2013], é proposto o uso de ontologias a fim de melhorar o desempenho da análise de sentimentos no *Twitter*™.

Como pode ser notado, existem várias técnicas de análise de sentimentos, entretanto, o foco desse estudo está na utilização de modelos de aprendizado de máquina, juntamente com técnicas de extração de características, a fim de treinar e classificar um conjunto de opiniões, de acordo com o esquema exibido pela Figura 2.

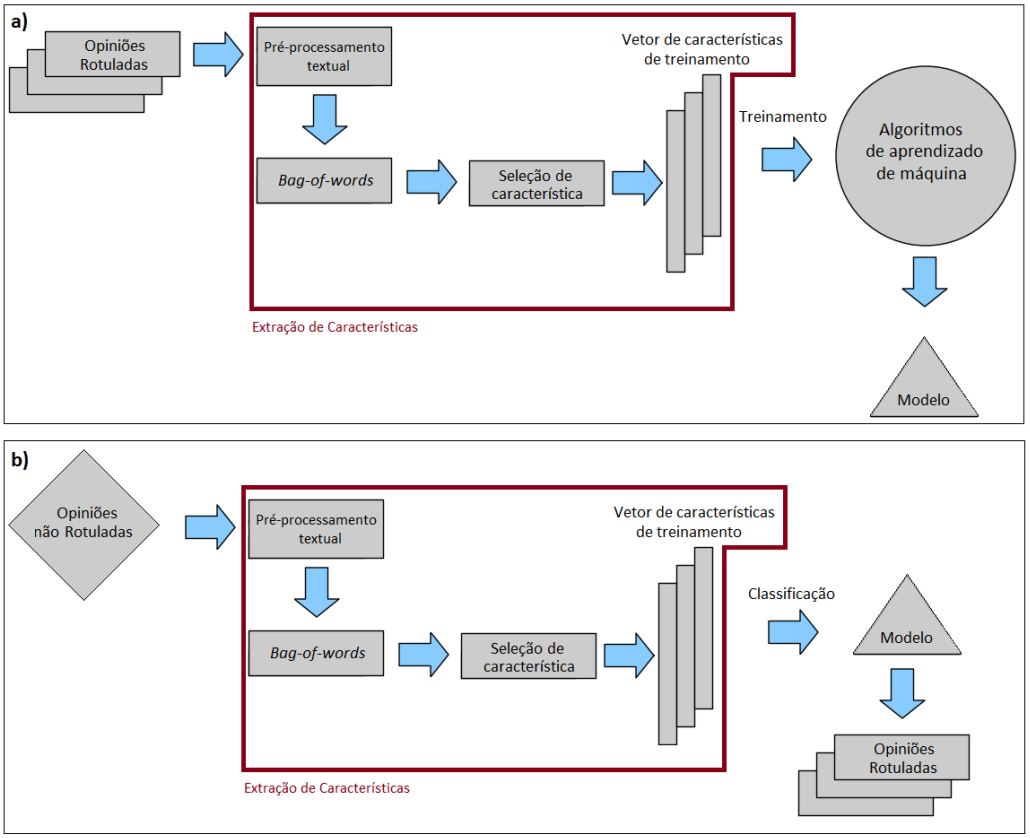

Figura 2. Processo de análise textual com extração de características e aprendizado de máquina. (a) Processo de treinamento. (b) Processo de classificação

Na parte a), o processo de extração de características e o treinamento dos algoritmos de aprendizado são descritos. Após a seleção de uma base de dados com opiniões previamente rotuladas, a fase de extração de características é dividida em quatro etapas. A primeira etapa, de pré-processamento textual, consiste na retirada de caracteres especiais, *stopwords* e tratamento da negação. A segunda etapa, *bag-of-words*, transforma cada opinião da base de dados em um conjunto de unigramas e bigramas. A terceira etapa é a fase de seleção de características que consiste na escolha dos melhores n-gramas para o treinamento dos algoritmos de classificação. A última etapa é a de vetorização que transforma a base de dados em documentos que são mais facilmente compreendidos pelos algoritmos de aprendizado. Por fim, esses algoritmos criam modelos de classificação que podem ser utilizados para categorizar opiniões sem rótulos.

A parte b) representa o modelo de classificação de novas instâncias. As opiniões não rotuladas selecionadas passam pelo mesmo processo de extração de características descrito na parte a). Após a extração de características, as opiniões são classificadas através de um modelo criado na parte a).

As etapas de pré-processamento, *bag-of-words*, seleção de características e vetorização de características para treinamento estão descritas na Seção 1.4. Essas técnicas estão presentes em alguns dos principais trabalhos que tem como objetivo a análise de sentimentos por meio do aprendizado de máquina. Na Seção 1.5, os principais algoritmos de classificação utilizados em análise de sentimentos são apresentados. Além disso, as medidas avaliativas são descritas na Seção 1.6.

Embora existam centenas de técnicas e métodos de análise de sentimentos presentes na literatura, como pode ser notado na Seção 1.7, que discute os trabalhos relacionados, alguns passos comuns e técnicas bem utilizadas foram selecionados baseado na importância dos trabalhos e no bom desempenho dos métodos existentes.

1.4 Extração de Características

Seja um conjunto *D* de opiniões selecionadas, algumas fases são essenciais na extração de características de textos que contêm opiniões. Embora as Técnicas de Extração de Características (TEC's) descritas a seguir não agreguem todas as formas de análise disponíveis, por meio destes passos é possível configurar um bom documento que possa ser compreendido por um algoritmo de aprendizado de máquina, cuja análise é o foco deste trabalho.

1.4.1 Pré-Processamento Textual

O primeiro passo para a construção de um documento compreensível para os algoritmos de classificação é selecionar uma base de dados com textos avaliativos, isto é, textos que possuam um sentimento em relação a um item.

Com as opiniões a serem analisadas devidamente selecionadas, o próximo passo do pré-processamento é a *tokenization*, que consiste na retirada de caracteres como vírgulas, acentos e pontuações. Em alguns trabalhos, alguns caracteres, como pontos de exclamação ou *emoticons* podem ser utilizados como característica de treinamento [GO; BHAYANI; HUANG 2009] ou como forma de seleção de opiniões para a criação de uma base de dados [PAK e PAROUBEK 2010]. Em casos nos quais as opiniões são extraídas diretamente de páginas web, a retirada de *tags* em HTML também deve ser realizada como feito em [BEINEKE *et al*. 2004] e [KANG; YOO; HAN 2012]. Alguns exemplos de caracteres especiais estão presentes na Tabela 1.

Tabela 1. Exemplos de caracteres especiais

Descrição	Token
Acentos	´ ~ ^
Pontuação	‘ ’ , . ; : ? !
Especiais	@ # * ( ) &
Emoticons	:) ;) :D :(:(;(
HTML	 <p>

Com a retirada desses caracteres especiais das opiniões, o próximo passo é o da normalização textual. Nesta etapa, estão incluídas a retirada de radicais, retirada de letras

repetidas em algumas palavras e a correção ortográfica. A etapa de correção ortográfica pode ser notada em trabalhos como [KOULOUMPIS; WILSON; MOORE 2011]. Embora seja indiscutível a necessidade desse passo, poucos trabalhos que tem o foco na classificação de sentimentos citam a normalização textual na fase de extração de características. Esses passos podem ser melhor estudadas em livros como [MANNING e RAGHAVAN 2009] que, além de apresentarem uma boa introdução sobre recuperação de informação e o pré-processamento textual, mostram a utilização de algoritmos de aprendizado para a classificação textual.

Outro passo importante na parte de tratamento das opiniões é a retirada de palavras consideradas com pouco ou nenhum sentimento, as chamadas *stopwords*[11]. O objetivo é diminuir a quantidade de palavras que possam ser usadas no treinamento, retirando palavras que pouco influenciam na determinação do sentimento final de um texto.

Outro passo importante é o tratamento de opiniões com palavras que expressam negação [PANG e LEE 2008]. Desta forma, frases como "*This is not bad*" ou "*That is not good*" tem seu sentimento invertido pelo *token* "*not*". A fim de tratar esse problema, palavras que tem como precedentes os modificadores *no, not ou nothing* podem ser transformadas em uma única palavra. Como exemplo, "*not good*" é representado pelo *token "not_good"* que é similar ao *token* "*bad"*.

1.4.2 N-Gramas – *Bag of Words*

Com as opiniões normalizadas, cada palavra de uma opinião corresponde a um unigrama, como pode ser observado no trabalho de Pang *et al.*[PANG; LEE; VAITHYANATHAN 2002]. Além de unigramas, essas palavras podem ser agrupadas formando bigramas (duas palavras) ou n-gramas (duas ou mais palavras). Unigramas e bigramas são as principais formas de representação de *tokens* e possuem bons resultados na análise de sentimentos [LIU 2012], tanto na classificação binária [PANG; LEE; VAITHYANATHAN 2002] como multiclasse [PANG e LEE 2005].

Seja a frase "*This cell phone is amazing*". Na Tabela 2, é exibido um exemplo da representação desta frase em unigramas e bigramas, sem que haja a retirada de nenhuma das palavras em etapas anteriores. Cada n-grama está separado por vírgulas na tabela e a união dos mesmos está representada pelo caractere "_". Nota-se que a ordem das palavras foi mantida em relação à estrutura da frase inicial e nem todos os n-gramas possíveis estão representados.

Tabela 2. Exemplo de bag-of-words com n-gramas

Unigrama	This, cell, phone, is, amazing
Bigrama	This_cell, cell_phone, phone_is, is_amazing

1.4.3 Técnicas de Seleção de Características

Após a etapa de normalização textual, a fase de Seleção de Características é fundamental para a escolha dos n-gramas para o treinamento de algoritmos de aprendizado [LIU 2012]. Como demonstrado por [PRUSA; KHOSHGOFTAAR; DITTMAN 2015] na análise de dados recolhidos do *Twitter*™[12], a seleção de características pode melhorar

[11] Lista de *stopwords* que será utilizada: http://www.ranks.nl/stopwords
[12] http://www.twitter.com

significativamente o desempenho da classificação. Esta etapa consiste na escolha de n-gramas que serão utilizadas como atributos de treinamento.

Três métodos de seleção de características foram testadas e analisadas: Information Gain, Gain Ratio e Chi-quadrado e estão descritas nas subseções posteriores. Trabalhos como [TANG; TAN; CHENG 2009], [SHARMA e DEY 2012] e [PRUSA; KHOSHGOFTAAR; DITTMAN 2015] fazem uso de alguma técnica de extração de característica.

1.4.3.1 Ganho de Informação

O ganho de informação é uma redução esperada na entropia causada pela divisão dos exemplos de acordo com um atributo qualquer *x*, na qual entropia é definida como o valor esperado de uma informação [HARRINGTON 2012], considerando-se *z* o número de classes possíveis que uma informação pode assumir dada pela Equação 1.1:

$$H = \sum_{i=1}^{z} p(x_i) \log_2 p(x_i) \quad (1.1).$$

Ele mede o número de bits obtidos por meio da predição de uma classe através da presença ou falta de um termo em um documento. Seja *t* um n-grama, o ganho de informação de um termo é calculado como na Equação 1.2:

$$IG(t) = -\sum_{i=1}^{z} P(c_i) \log P(c_i) + P(t) \sum_{i=1}^{z} P(c_i|t) \log P(c_i|t) + P(\bar{t}) \sum_{i=1}^{z} P(c_i|\bar{t}) \log P(c_i|\bar{t}) \quad (1.2),$$

onde $P(c_i)$ denota a probabilidade de uma classe *i* ocorrer; P(t) é a probabilidade de um n-grama (atributo) *t* ocorrer; e $P(\bar{t})$ a é a probabilidade de um n-grama *t* não ocorrer [TAN e ZHANG 2008].

Em análise de sentimentos, dado um conjunto de n-gramas de uma base de dados com opiniões, na qual duas classes (positivo ou negativo) existem, o IG para cada *token* é calculado com base na Equação 2. Para o problema de RIP com 5 classes, *i* varia de 1 a 5.

De acordo com a metodologia utilizada, apenas alguns n-gramas são utilizados para treinamento. Estes são escolhidos de acordo com a maior variação do ganho de informação, tanto para classes negativas quanto para classes positivas, isto é, palavras que expressam sentimento negativo, por exemplo, tem maior tendência a serem utilizadas em opiniões nas quais o autor não recomendaria um item.

1.4.3.2 Ganho de Médio de Informação

O ganho médio de informação aprimora o resultado do ganho de informação normalizando a contribuição de todas as características na decisão da classificação final para um documento. Na Equação 1.3, os valores de normalização ou *Split Information* são calculados por meio da informação obtida pela divisão de um documento de treinamento P em *v* partes, na qual *v* corresponde a um atributo *x* [SHARMA, A.; DEY 2012]:

$$SplitInfo(t) = -\sum_{j=1}^{v} \frac{|P_j|}{|P|} \log \frac{|P_j|}{|P|} \quad (1.3).$$

Por fim, a Equação 1.4 define o ganho médio como:

Gain Ratio(t) = *Information Gain*(t)/*SplitInfo*(t) (1.4).

Assim como no IG, essa fórmula tem como objetivo selecionar palavras que possuem algum sentimento, seja ele positivo ou negativo, e os n-gramas com maior ganho médio são utilizados como atributos.

1.4.3.3 Chi-Quadrado

Este modelo consiste em retirar os n-gramas mais comuns ou os que sejam mais próximos de palavras como "bom" ou "ruim" de um texto. A partir disso, vetores podem ser criados com palavras separadas (unigramas), duas palavras (bigramas) ou n-gramas.

Ele representa a associação entre uma característica e a classe correspondente por meio da Equação 1.5:

$$\mathrm{CHI}(t, c_i) = \frac{\mathrm{N}*(\mathrm{AD}-\mathrm{BE})^2}{(\mathrm{A}+\mathrm{E})*(\mathrm{B}+\mathrm{D})*(\mathrm{A}+\mathrm{B})*(\mathrm{E}+\mathrm{D})} \text{ and } \mathrm{CHI}_{\max} = \max_{\mathrm{i}}(\mathrm{CHI}(t, c_i)), \text{ (1.5),}$$

onde *t* é um n-grama e c_i a classe. A é o número de vezes que *t* e c_i ocorrem simultaneamente; B é o número de vezes que *t* ocorre sem c_i; E é o número de vezes que c_i ocorre sem *t*; D é o número de vezes que nem c_i nem *t* ocorrem e; N é o total de documentos [TAN e ZHANG 2008].

Para cada classe, a associação entre um atributo e uma classe é calculada, entretanto, apenas o valor máximo $\mathrm{CHI}_{\max}$ é utilizado, selecionando a classe com maior relação. Na análise textual, *t* é representado por um n-grama e *c* são as classes positivo ou negativo na classificação binária ou são as classes referentes as estrelas presentes no problema de inferência de ratings.

1.4.4 Vetorização

Com os n-gramas selecionados pelos métodos de extração de características citados na subseção anterior, a próxima etapa consiste em transformar uma frase em um vetor de características, onde os atributos correspondem aos n-gramas selecionados. Estes atributos são configurados de acordo com frequência dos mesmos em relação a uma opinião. Como exemplo, podemos notar o texto a seguir:

Opinião: *Great Hotel, lovely staff, great location.8 of us stayed here for 2 nights on a hen party, hotel is close to all bars night clubs, shopping, would definitely stay here again.Hotel is clean and security is great, rooms are really nice and comfortable and have great tv, kitchenette is very handy*. *Rating*: 5.
Words (15): *great, lovely, worst, location, stay, close, shop, terrible, clean, security, nice, comfortable, handy, bad, good.*
Matriz de representação

4	1	0	1	2	1	1	0	1	1	1	1	0	0	**5**

Nesse caso, cada posição do vetor corresponde exclusivamente a uma palavra e seu valor é dado pela frequência em uma determinada opinião. As *words* utilizadas acima são apenas exemplos, mas em uma aplicação real, essas palavras são selecionadas pelas técnicas de seleção de características citadas na Seção 1.4.3.

Utilizando o exemplo acima, notamos que a palavra *great* está na primeira posição do vetor. Sua frequência é dada pelo número de vezes que a palavra aparece na frase,

neste caso o número 4. A última posição do vetor corresponde à classe inicial (*rating*) da opinião. Toda a base de dados deve ser configurada seguindo este modelo a fim de criar um grande grupo de exemplos para o treinamento dos algoritmos de aprendizado de máquina supervisionados.

Além da frequência, outro valor pode ser utilizado para preencher cada posição do vetor. O modelo TF-IDF configura os vetores com um peso w_t para um termo *t* de acordo com a Equação 1.6:

$$w_t = f_t . idf_t = f_t . \log \frac{N}{df_t} \qquad (1.6),$$

onde f_t é o número de vezes que *t* ocorre em uma opinião *d;* idf_t é a frequência inversa em um documento do termo *t; N* é o total de opiniões e df_t é o número de opiniões que contém *t* [PALTOGLOU e THELWALL 2010]. Além do trabalho de Paltoglou e Thelwall, que testa várias variantes deste modelo, esta fórmula de representação apresenta bons resultados no trabalho de Martineau e Finin [MARTINEAU e FININ 2009].

Opinião: *Great Hotel, lovely staff, great location.8 of us stayed here for 2 nights on a hen party, hotel is close to all bars night clubs, shopping, would definitely stay here again.Hotel is clean and security is great, rooms are really nice and comfortable and have great tv, kitchenette is very handy*.
Rating: 5.
Words (15): *great, lovely, worst, location, stay, close, shop, terrible, clean, security, nice, comfortable, handy, bad, good.*
Matriz de representação

0.887	0.2342	0	0.231	0.887	0.121	0.164	0	0.164	0.123	0.2342	0.421	0	0	**5**

1.5 Modelos e Algoritmos de Classificação

Com todo o processo de seleção de características finalizado, criando, por fim, arquivos com atributos quantitativos que são mais facilmente compreendidos e executados por algoritmos de aprendizado, a próxima seção apresenta alguns dos principais modelos e algoritmos nativos utilizados em análise de sentimento para resolver problemas multiclasse. Além disso, ela apresenta dois métodos que utilizam uma forma de classificação binária: *one-versus-one* (OvO) e o *one-versus-all* (OvA), métodos conhecidos como multiclasse adaptado. Os trabalhos que utilizam alguns desses algoritmos estão bem descritos em [LUNARDI; VITERBO; BERNARDINI 2015] e na Seção 1.7 desse trabalho.

1.5.1 Naive Bayes

O algoritmo Naive Bayes é uma variação da teoria de decisão Bayesiana. A probabilidade Bayesiana habilita o conhecimento inicial e a lógica a serem aplicados em declarações desconhecidas [HARRINGTON 2012]. Formalmente, pode-se calcular a probabilidade condicional como na Equação 1.7:

$$P(c \mid d) = \frac{P(c)P(d \mid c)}{P(d)} \qquad (1.7)$$

Uma variação a teoria bayesiana, o modelo multinomial captura a frequência de uma palavra no conjunto de opiniões [MCCALLUM e NIGAM 1998]. Para associar a

um novo exemplo *t* uma classe c_i, a classe com maior probabilidade $c^* = \text{argmax}\, P(c_i|t)$ é considerada. Na Equação 1.8 é mostrado como o cálculo das probabilidades para cada classe $c_i \in c$ é realizado.

$$P_{NB}(c_i|t) = P(c_i)(\prod_{j=1}^{D} P(t_j|c_i) \qquad (1.8),$$

onde *t* é um termo, *i* é o número da classe e *D* é o conjunto de opiniões.

A partir de um conjunto de termos *t* de uma opinião, representado pelo vetor *w*, a distribuição das probabilidades é dada pela Equação1. 9:

$$P_{NB}(c_i|w) = P_{NB}(c_i|t_1..t_n) = \frac{P(w|c_i)P(c_i)}{P(w)} \qquad (1.9),$$

no qual *n* é o número de termos em um vetor *w*.

Um dos trabalhos iniciais de análise de sentimentos [PANG; LEE; VAITHYANATHAN 2002] utiliza, além do SVM e da Entropia Máxima (EntMax), o Naive Bayes já que este demonstrava bons resultados no problema de categorização de textos. Apesar de simples, o Naive Bayes apresentou bons resultados, superando os outros algoritmos quando treinado com unigramas. Para o problema de multiclasse, Long et. al [LONG; ZHANG; ZHUT 2010] utilizam tanto um modelo de regressão quanto o Naive Bayes, sendo estes treinados com características retiradas de opiniões com uma técnica baseada na complexidade Kolmogorov. Assim como em [PANG; LEE; VAITHYANATHAN 2002], o resultado é satisfatório, chegando a atingir cerca de 12,5% de melhoria de desempenho com os classificadores testados em relação aos trabalhos anteriores.

1.5.2 SVM

Dado um conjunto de dados linearmente separável, caso exista uma linha em um plano que possa separar o conjunto de dados, a linha é chamada de hiperplano separador. A ideia é encontrar o hiperplano que esteja o mais próximo possível dos pontos, sendo que esses pontos estejam o mais distante possível do hiperplano a fim de garantir a melhor robustez do classificador. Isso é chamado de margem. Os pontos mais próximos da margem são chamados de vetores de suporte [HARRINGTON 2012], como pode ser visto na Figura 3.

A ideia principal do modelo de máquina de vetores de suporte é encontrar as margens ótimas em relação a um hiperplano separador *h*. Essa distância é calculada pela fórmula $u = \vec{w}.\vec{x} - b$, no qual $\vec{w}$ é o vetor normal para o hiperplano, $\vec{x}$ é o vetor de entrada e b é uma constante.

Para o caso linear, a margem é definida pela distância do hiperplano em relação ao vizinho mais próximo dos exemplos positivos e negativos. Maximizar esta margem pode ser expresso por um problema de otimização, no qual a maximização $\frac{2}{||\vec{w}||^2}$ é equivalente a minimizar o problema, conforme a Equação 1.10:

$$L(w) = \frac{||\vec{w}||^2}{2} \quad (1.10),$$

sujeito a $y_n(\vec{w}.\vec{x} - b) \geq 1$, $\forall n$ no qual x_n é o *n*-ésimo exemplo de treinamento e y_n é a saída correta do SVM para o *n*-ésimo exemplo de treinamento.

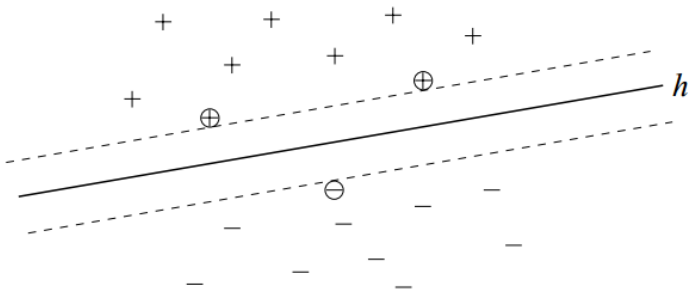

Figura 3. Hiperplano *h* encontrado, separando dados de treinamento positivos e negativos. Dados circulados são vetores de suporte [JOACHIMS 1998]

Para problemas com a margem suave, variáveis de relaxamento são utilizadas para flexibilizar as restrições do problema de otimização descrito na Equação 1.10. Essas variáveis ξ medem o local de uma amostra em relação as margens. Nesse caso, a Equação 1.10 fica sujeita a $y_n(\vec{w}.\vec{x} - b) \geq 1 - \xi$.

Para problemas não lineares, nem sempre é possível encontrar um hiperplano H para o problema. Para esse tipo de problema, é preciso encontrar uma transformação φ(x) que não seja linear, de acordo com a Equação 1.11.

$$\varphi(x) = \varphi_1(x), ..., \varphi_m(x) \quad (1.11),$$

dado *m* o número de dimensões do problema. Nesse caso, os padrões $\vec{x}$ passam a ser linearmente separáveis e o SVM fica sujeito as restrições $y_n(\vec{w}.\varphi(\vec{x}) - b) \geq 1$. Para um conjunto de *n* padrões $\varphi(\vec{x}_n)$, multiplicadores de Lagrange podem ser utilizados. A solução depende apenas do produto $\varphi(\vec{x}_i)\varphi(\vec{x}_j)$, que pode ser obtido por meio de funções conhecidas como Kernels, como o polinomial mostrado na Equação 1.12.

$$K(\vec{x}_i, \vec{x}_j) = (\delta(\vec{x}_i.\vec{x}_j) + k)^d \text{ (1.12).}$$

Para problemas multiclasses, são necessários vários classificadores binários que podem ser construídos por meio de técnicas adaptadas, descritas na Seção 1.5.5. Em muitos trabalhos, algumas variantes deste modelo são utilizadas. Isso pode ser notado em [BROOKE 2009] e em [PANG e LEE 2005], no qual o algoritmo Sequential Minimal Optimization (SMO) é o mais indicado para resolver o problema de análise de sentimentos multiclasse já que ele é utilizado para resolver problemas de regressão a partir do SVM. O SMO divide o problema de programação quadrática (PQ) existente no SVM simples, criando soluções menores para o problema de PQ sem utilizar uma matriz de armazenamento extra [PLATT 1998]. Além do SMO, variações do pacote LibSVM[13] com a função linear sendo utilizada podem ser adaptadas para a classificação multiclasse. Isto se deve ao fato de que, segundo [DUMAIS *et al*. 1998] e [KAESTNER, 2013], o modelo linear é o mais adequado para análise de texto.

Assim como dito em [LIU, 2012], o problema de inferência de *rating* também pode ser considerado um problema de regressão. Isso faz com que variantes do SVM, como o SMO, estejam presentes em trabalhos como [PANG e LEE 2005], [LONG; ZHANG; ZHUT 2010] e [DE ALBORNOZ *et al*. 2011], trabalhos estes que possuem

[13] https://www.csie.ntu.edu.tw/~cjlin/libsvm

bons resultados e são referências na área de inferência de *rating* ou classificação multiclasse.

1.5.3 K-NEAREST NEIGHBORS (KNN)

O k-Nearest Neighbors (kNN) é um método baseado em instâncias que aprende com o simples armazenamento dos dados de treinamento. Quando uma nova instância surge, ele recupera os dados armazenados e classifica essa nova instância [MITCHELL 1997]. A partir dos k vizinhos mais parecidos, ele escolhe o dado com os k mais similares com o que será classificado e atribui uma nova classe a ele [HARRINGTON 2012]. A proximidade dos vizinhos pode ser definida, por exemplo, de acordo com a distância Euclidiana [MITCHELL 1997] demonstrada na Equação 1.13 para dois vizinhos:

$$u = \sqrt{(xA_0 - xB_0)^2 + (xA_1 - xB_1)^2} \text{ (1.13).}$$

Em Tan e Zhang [TAN e ZHANG 2008], considerando *d* um documento de teste, a tarefa está em encontrar os *k* vizinhos entre os outros documentos de treinamento. Na Equação 1.14, a similaridade entre o item *d* e os outros vizinhos é usada como o peso das classes dos documentos mais próximos, calculado como:

$$\text{score(d,}c_i\text{)}=\sum_{d_j \in KNN(d)} sim(d, d_j)\delta(d_j, c_i) \qquad (1.14),$$

no qual KNN(*d*) representa o conjunto de vizinhos do documento *d* e c_i uma classe. A função sim(d, d_j) representa a similaridade entre um documento *d* o documento de treino d_j. Se d_j pertence a c_i, $\delta(d_j, c_i)$ é igual a 1, senão, é igual a 0. Logo, o documento *d* deve pertencer à classe que ele possui o maior *score*.

Além de Tan e Zhang, entre os trabalhos relacionados, apenas em Sharma e Dey [SHARMA e DEY 2012] o kNN é utilizado para o problema de classificação de sentimentos binária. Em ambos trabalhos, o kNN apresenta resultado bem inferior quando comparado com os algoritmos SVM e Naive Bayes. Para o RIP, em nenhum dos trabalhos citados nesta pesquisa foi utilizado este algoritmo, o que serviu como motivação para avaliar o desempenho do mesmo neste trabalho. Na ferramenta utilizada neste estudo, o algoritmo kNN é conhecido como IBk (Instance-Based Learning with Paramater k).

1.5.4 Árvores de Decisão

As árvores de decisão são um dos principais métodos de inferência indutiva utilizadas. Elas consistem em um método de aproximação discreta do alvo, na qual a função de aprendizado é representada por uma árvore de decisão, que podem ser representadas como um conjunto de regras *if-then* [MITCHELL 1997].

A tarefa de construir uma árvore de indução consiste em criar uma regra de classificação que pode determinar a classe de objeto a partir dos valores dos seus atributos. Essa regra de classificação é ser expressa por meio de uma árvore de decisões. As folhas de uma árvore são as classes existentes do problema e os nós internos são os atributos escolhidos no treinamento. A classificação de um novo objeto começa na raiz e para cada atributo uma decisão é tomada a fim de chegar em um novo atributo. Esse processo continua até que a classe apropriada seja encontrada [QUINLAN 1986].

Para o problema de análise de sentimentos multiclasse, os atributos dos nós internos são as características de treinamento selecionados pelos métodos de seleção da Seção 1.4.3 e as folhas representam as classes (para o problema de inferência de *ratings*, cada folha é o valor numérico das estrelas).

O algoritmo ID3, uma implementação de uma árvore de decisões, foi criado para problema nos quais existem muitos atributos e o conjunto de treinamento possui vários exemplos. A ideia básica deste algoritmo é iterativa. Um subconjunto de treinamento é escolhido aleatoriamente e uma árvore é criada a partir dele. Para o restante dos objetos de treinamento são classificados por meio da árvore inicial. Se o restante do conjunto for corretamente classificado, o processo de construção é finalizado. Senão, um conjunto de objetos que não foram corretamente classificados são adicionados ao subconjunto inicial e uma nova árvore é criada. Esse processo pode ser finalizado por meio de um limite de iterações ou até que uma árvore classifique todos os dados de treinamento corretamente. Além do ID3, a variação C4.5 [SHARMA e DEY 2012] é utilizada em análise de sentimentos.

Na pesquisa de Albornoz *et al.* [DE ALBORNOZ *et al*. 2011], um modelo de árvore (Functional Tree - FT) é utilizado a fim avaliar o vetor de intensidade de características proposto pelos autores, juntamente com o LibSVM e o algoritmo Logistic. Eles utilizam estes algoritmos com o intuito de prever o *rating* final de uma opinião, atingindo a acurácia de 43,7% para o modelo FT. Entre todos os algoritmos utilizados, a FT obteve o pior desempenho, sendo 3,2% inferior ao modelo Logistic (46,9%).

Chen *et al.* [CHEN *et al*. 2006] criam uma análise visual de opiniões sobre o livro *O Código da Vinci* inspirados em um modelo semelhante a uma árvore de decisões. Além disso, eles utilizam os algoritmos C4.5, SVM e Naive Bayes a fim de selecionar bons termos para a categorização das opiniões utilizadas.

1.5.5 Algoritmos Multiclasse Adaptados

Nessa seção, dois dos principais métodos para resolver problemas multiclasse por meio de divisões binárias são descritos: o *One-vs-One* (OvO) e o *One-vs-All* (OvA). O método OvA cria *n* divisões, na qual cada etapa do aprendizado é feito comparando um a classe a todas as outras classes. No modelo OvO, cada classe c_i é comparada com outra classe c_k, onde *k*, *i* = 1..*n* e $i \neq k$, dado que *n* é o número de classes [HSU e LIN 2002].

Em um modelo OvA, a partir da escolha de um classificador (SVM, por exemplo), *n* classificadores são construídos, isto é, para cada comparação entre uma classe e as demais, um classificador é construído. Para o *i*-ésimo classificador, os exemplos positivos são todos os pontos da classe *i* e os exemplos negativos são todos os pontos que não estão na classe *i*. Seja f_i o *i*-ésimo classificador, a classificação é dada por meio da Equação 1.15:

$$f(x) = \arg\max_i f_i(x) \quad (1.15).$$

Para o classificador OvO, um modelo classificador também é escolhido, entretanto, cada classe *j* é comparada com outra classe *i*. Seja f_{ij} o classificador no qual as classes i são exemplos positivos e as classes *j* são exemplo negativos. Assumindo que $f_{ij} = -f_{ij}$, a classificação será feita por meio da Equação 1.16:

$$f(x) = \arg\max_{i} \left(\sum_j f_{ij}(x)\right). \quad (1.16).$$

Em relação ao número de divisões binárias necessárias em cada um desses métodos, no classificador *one-vs-all* ele é dado por *i*, onde *i=n*, sendo *n* o número de classes. Já para o algoritmo *one-vs-one*, o número de etapas para a classificação é dado por $\frac{n(n-1)}{2}$, onde *n* é o número de classes [ALY 2005]. Para o problema multiclasse, os algoritmos SVM citados acima (SMO e LibSVM) utilizam o modelo OvO.

Para um problema com 4 classes {1, 2, 3, 4}, o OvO cria 6 classificadores (1-2, 1-3, 1-4, 2-3, 2-4, 3-4). Para a criação de cada classificador binário, as instâncias de treinamento possuem o rótulo correspondente a cada classificador, isto é, para uma divisão binária 3-4, apenas exemplos de treinamento classificados como 3 ou 4 são utilizados.

Na fase de classificação, a classe escolhida é baseada em uma votação direta dada pelo maior valor de acordo com a Equação 1.16, selecionando a classe com maior número de votos. Exemplificando, dado uma nova instância *a*, a Tabela 3 e a Tabela 4 mostram uma predição para esse novo dado.

Tabela 3. Votos de cada classificador do modelo OvO

Classificador	f(*a*)=
1-2	2
1-3	1
1-4	1
2-3	2
2-4	2
3-4	3

Tabela 4. Contagem dos votos para cada classe

	Votos para cada classe			
Classe	1	2	3	4
Número de votos	2	**3**	1	0

Nesse exemplo, a classe 2 é a escolhida como rótulo da nova instância a. Em caso de empata, a escolhe é feita aleatoriamente [PIMENTA 2004].

Avaliando o modelo OvA para o mesmo número de classes, 4 classificadores são criados (1 vs {234}, 2 vs {134}, 3 vs {124}, 4 vs {123}). Nesse caso, na fase de treinamento, todas as classes são utilizadas em cada divisão. No processo de classificação, dada uma nova instância *a*, a predição é dada por meio de Equação 1.15, utilizando uma votação direta distribuída, conforme exibido na Tabela 5. Nesse exemplo, a classe 2 recebe o maior valor (1,999). Dessa forma, a instância *a* é classificada como 2.

Estas abordagens são comumente utilizadas quando algoritmos SVM são indicados para o problema, por exemplo. Para a avaliação multiclasse, os algoritmos SVM citados acima (SMO e LibSVM) utilizam o modelo de divisões OvO.

Tabela 5. Votos de cada classificador do modelo OvA

f(*a*)		Votos			
Classificador		1	2	3	4

1 *vs* {234}	1	0	0,333	0,333	0,333
2 vs {134}	Outra	0	1	0	0
3 vs {124}	Outra	0,333	0,333	0	0,333
4 vs {123}	Outra	0,333	0,333	0,333	0
Total		0,666	**1,999**	0,666	0,666

1.6 Avaliação de Desempenho

Para medir o desempenho dos algoritmos e técnicas citados anteriormente, tipicamente são utilizadas medidas avaliativas, que se baseiam na matriz de confusão. Essas medidas são as mais utilizadas em outros trabalhos como [PANG; LEE; VAITHYANATHAN 2002], [TAN e ZHANG 2008], [GOLDBERG e ZHU 2006] e [GO; BHAYANI; HUANG 2009], seja para a análise multiclasse ou binária.

Uma matriz de confusão para um problema *n*-classes é uma matriz *n* x *n* [GODBOLE e SARAWAGI 2004] onde o elemento M_{ij} é, para *i=j*, o número de opiniões pertencentes a uma classe *i* que foram corretamente classificadas e, para *i≠j*, o número de opiniões de uma classe *i* que foram erroneamente classificadas em outra classe *j*. Na Tabela 6 é apresentado um exemplo de uma matriz em que as letras *a*-*e* correspondem à escala de *ratings* utilizada (1-5 estrelas), respectivamente.

1.6.1 Acurácia, Precisão e Recall

Com base na matriz de confusão apresentada na Tabela 6, as medidas descritas a seguir são muito utilizadas a fim de medir o desempenho da precisão dos algoritmos, principalmente a acurácia do modelo que mede quão *T* se aproxima de *S*, onde *T* é o conjunto inicial e *S* é o conjunto com as predições criadas para uma base de dados.

Tabela 6. Exemplo de uma matriz de confusão para o problema 5-classes

a	b	c	d	e	Total	← classificado como
1140	276	61	10	13	1500	a=1
497	502	380	96	25	1500	b=2
132	260	773	281	54	1500	c=3
47	97	228	648	480	1500	d=4
19	36	45	267	1133	1500	e=5
1835	**1171**	**1487**	**1302**	**1705**	7500	

A acurácia é calculada como:

$$\mathrm{A} = \frac{número\ de\ exemplos\ classificados\ corretamente}{total\ de\ exemplos}$$

Analisando a Tabela 6, a acurácia final é dada pelo número de exemplos corretamente classificados (1140+502+773+648+1133) dividido pelo número total de exemplos (7500). Desta forma, a acurácia é dada por 0,5594.

A precisão é dada por:

$$\mathrm{P} = \frac{número\ de\ corretas\ predições\ positivas}{número\ de\ predições\ positivas}$$

Analisando a Tabela 6, a precisão para a classe *a* é dada pelo número de corretas predições positivas (1140) dividido pelo número de predições positivas (1835), isto é, o

número de objetos classificados como *a* e que inicialmente eram rotulados como *a* dividido pelo número de exemplos classificados como *a*, sejam eles inicialmente *a* ou não. Desta forma, a precisão é dada pelo valor 0,6212.

O *recall* é dado pela seguinte fórmula:

$$R = \frac{número\ de\ corretas\ predições\ positivas}{número\ de\ exemplos\ positivos}$$

Analisando a Tabela 6, o *recall* para a classe *a* é dada pelo número de corretas predições positivas (1140) dividido pelo número de exemplos positivos (1500), isto é, o número de objetos classificados como *a* e que inicialmente eram rotulados como *a* dividido pelo número de exemplos inicialmente rotulados como *a*. Desta forma, a precisão é dada pelo valor 0,76.

1.6.2 Acurácia Aproximada

O cálculo da acurácia aproximada é definido por Brooke [BROOKE 2009], e esta medida considera aceitável quando uma opinião é classificada com a classe exata ou com a(s) classe(s) vizinhas à classe exata, considerando a escala de *ratings* (1 a 5). A Tabela 7 estende a Tabela 6 para incluir os valores da acurácia aproximada para cada classe. Analisando a classe *b*, tanto opiniões classificadas como *a* ou *c* são aceitáveis e as opiniões inicialmente rotuladas como *b* e classificadas como *d* e *e* são consideradas como erro. Desta forma, se notarmos a acurácia exata da classe *b* (0,335) e a acurácia próxima, concluímos que muitas opiniões da classe *b* foram classificadas como *a* (497) ou *c* (380). Logicamente, o valor da acurácia aproximada é sempre mais elevado do que a acurácia exata.

Tabela 7. Matriz de confusão com acurácia exata e próxima

a	b	c	d	e	Acurácia		← classificado como
					Exata	Aproximada	
1140	276	61	10	13	0,76	0,944	a=1
497	502	380	96	25	0,335	0,919	b=2
132	260	773	281	54	0,515	0,876	c=3
47	97	228	648	480	0,432	0,904	d=4
19	36	45	267	1133	0,755	0,933	e=5
1835	**1171**	**1487**	**1302**	**1705**	**0,559**	**0,915**	-

Essa medida avaliativa também foi utilizada em [PALTOGLOU e THELWALL 2013], no qual os autores consideram não só a acurácia e o erro quadrático médio, mas também exibem o valor do erro absoluto médio e a acurácia aproximada, onde a distância máxima analisada é de uma classe para a classe correta.

1.7 Pesquisas em Análise de Sentimentos

Como foi discutido na seção anterior, os principais métodos de análise de sentimentos podem ser divididos em quatro grandes áreas, de acordo com um modelo semelhante ao de [CAMBRIA *et al.*, 2013]:

- afinidade léxica;
- aprendizado de máquina;
- orientação semântica, e;

- conceitos ou ontologias.

O aprendizado de máquina ou métodos estatísticos consistem na utilização de algoritmos como Naive Bayes e Máquina de Vetores de Suporte a fim de treinar um corpo textual e, a partir do treinamento, classificar novas opiniões. Esses métodos foram anteriormente abordados na Seção 1.5, já que este trabalho tem como foco a proposta de uma técnica que utilize estes algoritmos na análise de sentimentos. Desta forma, esta seção apresenta uma discussão dos principais trabalhos que utilizam algoritmos de classificação para a análise de sentimentos.

Estes trabalhos foram escolhidos com base na importância dos mesmos para a área de análise de sentimentos, levando em consideração os resultados obtidos e as técnicas de extração e algoritmos utilizados. Em alguns casos, algumas destas técnicas e algoritmos não foram citados na seção anterior devido ao grande número de técnicas disponíveis, sendo inviável que todas sejam descritas. Em relação às formas de utilização das opiniões, os principais trabalhos se distribuem em três campos que merecem destaque:

- a classificação em relação à objetividade ou subjetividade;
- a classificação binária, e;
- a classificação multiclasse.

Esses campos serão descritos nas seções abaixo, com destaque para os trabalhos de classificação binária e multiclasse.

1.7.1 Classificação em Texto Objetivo ou Subjetivo

A primeira etapa para realizar a classificação de textos é saber se eles são subjetivos, isto é, contém algum tipo de opinião em relação a uma entidade. Desta forma, tendo uma base de dados que não garanta que existam apenas textos com opiniões subjetivas, uma primeira etapa a ser realizada no processo de análise de sentimentos deve ser separar tais textos em relação à objetividade ou subjetividade. Wiebe e Rillof [WIEBE e RILLOF 2005] desenvolveram um classificador subjetivo usando textos não rotulados para o treinamento. A pesquisa inicia com um processo de busca que utiliza um dicionário de palavras subjetivas para criar os dados de treinamento automaticamente. Esses dados são utilizados para criar um modelo de extração de características e um classificador probabilístico. Finalmente, eles adicionam um mecanismo de autotreinamento que providencia um auxílio aos classificadores, enquanto eles ainda dependem de dados não anotados.

Yu e Hatzivassiloglou [YU e HATZIVASSILOGLOU 2003] utilizaram a similaridade entre sentenças e um classificador Naive Bayes para classificar um texto como subjetivo ou objetivo, baseando-se na afirmativa de que opiniões são mais similares a outras opiniões do que a textos factuais. Eles utilizaram um sistema chamado SIMFINDER para medir a similaridade entre as palavras e frases utilizadas nas diversas sentenças de treinamento. Para realizar a classificação final (objetivo ou subjetivo), os autores utilizaram técnicas de extração como n-gramas, marcadores POS e palavras que possuam algum sentimento. Além disso, a proposta também realizou a classificação binária de uma sentença classificada como subjetiva.

1.7.2 Classificação Binária

Muitos dos trabalhos existentes na área de análise de sentimentos têm como principal objetivo avaliar o desempenho de um ou mais algoritmos de aprendizado, comparando o resultado final, seja por meio da acurácia, tempo ou outras medidas avaliativas. Para isso, são utilizadas bases de dados com avaliações disponíveis na web, com o intuito de avaliar os melhores algoritmos e as melhores técnicas de extração de características. O principal objetivo destes estudos é a classificação em relação à polaridade de uma opinião, isto é, saber se ela é negativa ou positiva; boa ou ruim; recomendada ou não recomendada.

Essa seção discute a grande maioria dos trabalhos referenciados em nossa pesquisa, muitos das quais serviram de base para a metodologia utilizada. Isso se deve ao fato de o problema de análise de sentimentos ser geralmente considerado como um problema de classificação binária [LIU 2012]

Pang *et al.* [PANG; LEE; VAITHYANATHAN 2002] tinham como principal objetivo determinar se uma avaliação é positiva ou negativa utilizando algoritmos de aprendizado. Os autores compararam o desempenho destes algoritmos no problema de mineração de opiniões com o desempenho na classificação feita por humanos e na categorização baseada em tópicos. Os autores mostraram que os algoritmos são melhores na classificação do que humanos, mas seu desempenho não é melhor do que tradicionais métodos de categorização baseado em tópicos (classificação por assunto). Eles utilizaram uma base de dados de avaliações de filmes e pediram para que dois estudantes criassem uma seleção de palavras que indicavam a positividade ou negatividade de uma avaliação. Baseado nessa lista, eles criaram novos vetores de palavras que serão utilizadas pelos algoritmos Naive Bayes, SVM e Entropia Máxima. O desempenho alcançado foi melhor do que as bases formadas por humanos, mas em relação à acurácia de 90% da categorização baseada em tópicos, nenhum dos algoritmos, mesmo quando combinados com bigramas, POS ou a posição de um n-grama no texto conseguiu atingir tal desempenho. O melhor classificador foi o SVM, enquanto a utilização de unigramas mostrou-se mais efetiva em relação às características.

Kang *et al.* [KANG; YOO; HAN 2012] propuseram um novo método para a análise de sentimentos de opiniões sobre restaurantes apresentando duas melhorias no algoritmo Naive Bayes a fim de resolver o problema de balanceamento das acurácias das classificações positivas e negativas. Eles combinaram técnicas de unigramas e bigramas (que incluem tratamento de palavras negativas e utilização de advérbios intensivos) com o algoritmo SVM, o Naive Bayes e as melhorias do Naive Bayes propostas pelos autores. Os autores demonstraram que o Naive Bayes proposto, quando implementado usando bigramas e unigramas, diminui a distância entre a acurácia positiva e a acurácia negativa para 3.6% comparada ao Naive Bayes original e em até 28% em relação ao SVM para opiniões sobre restaurantes.

Xia *et al.* [XIA; ZONG; LI 2011] fizeram um estudo sobre a efetividade do agrupamento de técnicas para tarefas de classificação binária, focando no agrupamento de conjuntos de características e algoritmos de classificação. Eles projetam dois esquemas utilizando POS e dependência sintática e, para cada esquema, utilizam NB, SVM e a Entropia Máxima para a classificação, utilizando a base de dados de filmes disponíveis

em Cornell[14] e o Multi-Domain Sentiment Dataset[15] com avaliações sobre produtos da *Amazon*™.

Tan e Zhang [TAN e ZHANG 2008] fizeram um trabalho que apresenta um estudo sobre análise de sentimentos que não usa a língua inglesa, mas sim a chinesa. Os autores utilizam quatro métodos de seleção de características (Informação Mútua, IG, DF e CHI) e cinco algoritmos de aprendizado de máquina (kNN, Naive Bayes, SVM, Winnow e o classificador centroide, estes dois últimos não citados neste estudo) em uma base de dados que contém opiniões sobre três domínios: educação, filmes e eletrodomésticos. Considerando todos os algoritmos de aprendizado, o melhor método de seleção de característica é o Ganho de Informação, que atinge uma média de 88.6% de acurácia. Considerando os métodos de seleção de características, em relação aos algoritmos de aprendizado, o SVM produz a melhor acurácia: 86.8%. Em um dos testes, os autores realizaram o treinamento do SVM em um domínio de eletrodomésticos e utilizaram o conhecimento adquirido para classificar opiniões no domínio de educação. Os autores surpreendentemente obtiveram 0,899 para o valor do MacroF1 para o SVM treinado, ilustrando a possibilidade do uso de modelos treinados em um domínio serem utilizado em outros.

Matsumoto *et al.* [MATSUMOTO *et al.* 2005] analisaram o desempenho do SVM para realizar a classificação binária de avaliações sobre filmes, utilizando dois conjuntos de dados. Os autores extraíram unigramas, bigramas, frequentes subsequências de palavras e sub-ávores dependentes, e usaram tais características para o treinamento de um classificador SVM. Entre os vários testes, eles atingiram 88.3% de acurácia para a primeira base de dados utilizando bigramas, unigramas e árvores de dependência, e 93.7% para o segundo conjunto, utilizando o SVM com bigramas, unigramas, palavras subsequentes e árvores de dependência.

Paltoglou and Thelwall [PALTOGLOU e THELWALL 2010] mostraram que funções de peso adaptadas da Recuperação de Informação (RI) baseadas no cálculo da *tf.idf* [25] e adaptadas para uma configuração particular da análise de sentimentos podem aumentar significativamente o desempenho da classificação. Os autores mostraram que a utilização do SVM adaptado como algoritmo de aprendizado e com essas funções de peso no processo de vetorização, os resultados atingiram até 96% de acurácia. Esse resultado está entre os melhores desempenhos entre os trabalhos relacionados para classificação binária utilizando um algoritmo de aprendizado.

Sharma e Dey [SHARMA e DEY 2012] exploraram cinco métodos de seleção de características em mineração de dados e sete algoritmos de aprendizado de máquina para análise de sentimento em um conjunto de avaliações on-line de filmes. Entre os melhores resultados, o método Gain Ratio (GR), uma variação de IG, foi o que apresentou os melhores resultados. Já em relação aos algoritmos de aprendizado, o SVM possuiu a melhor média de desempenho, considerando as cinco estratégias de seleção, mas o melhor resultado é apresentado pelo Naive Bayes atingindo 90,9% com GR.

Como pode ser observado, muitas das aplicações exploraram novas configurações e novos métodos para melhorar o desempenho dos algoritmos de aprendizado. Xia *et al.* [XIA *et al*. 2011] exploraram métodos agrupados: regras fixas e métodos treinados a fim

[14] Disponível em www.cs.cornell.edu/people/pabo/movie-review-data/.
[15] Disponível em www.cs.jhu.edu/~mdredze/datasets/sentiment/

de melhorar o desempenho dos algoritmos de aprendizado. Sharma and Dey [SHARMA e DEY 2012] fizeram um estudo sobre vários métodos de seleção de características e algoritmos de aprendizado. Paltoglou and Thelwall [PALTOGLOU e THELWALL 2010] utilizaram várias variações do inverso da frequência e atingem acurácia superior a 95%.

Pode ser notado também que alguns trabalhos utilizaram diversos algoritmos de aprendizado, combinados com diversas TEC's, mostrando que em muitos trabalhos houve algum tipo de comparação a fim de obter o melhor algoritmo para o(s) domínio(s) em estudo. Entre as principais TEC's destacaram-se as que analisaram termos e sua frequência em uma opinião. Entre os principais métodos de análise textual estão DF, IG, CHI, unigramas e n-gramas.

Os unigramas e n-gramas foram usados juntamente com outra técnica de extração em algumas pesquisas, com o intuito de selecionar os n-gramas mais importantes e calcular a frequência dos mesmos. Por exemplo, em [PAK e PAROUBEK 2010] foram usados n-gramas para representar palavras que foram obtidas através da análise da frequência de tais palavras chaves, além de marcadores POS. Em [PANG; LEE; VAITHYANATHAN 2002] foram utilizados unigramas, bigramas, POS e adjetivos, considerando em alguns casos a frequência, e em outras a presença de uma palavra. Em [19] foram testados cinco TEC's e sete algoritmos de aprendizado.

Embora exista um grande número de TEC's, esse estudo considera apenas os algoritmos e as TEC's mais utilizados, que foram descritos nas Seções 1.4 e 1.5. Esses métodos geralmente apresentaram bons resultados em outros trabalhos relacionados e aparecem em trabalhos de grande importância na área de análise de sentimentos utilizando algoritmos de aprendizado. Um resumo de todos esses trabalhos está presente na Tabela 8.

1.7.3 Classificação Multiclasse

Os problemas de classificação multiclasse agregam trabalhos que analisam problemas que podem ser divididos em 3 ou mais classes. O problema de inferência de *ratings* é considerado um problema multiclasse, seja em uma escala com 3, 4, 5 ou mais estrelas. Esses problemas também são conhecidos como problemas de escala de multiponto [PANG e LEE 2008].

Em [PANG e LEE 2005], os autores avaliaram a acurácia de humanos em relação à tarefa de determinar o *rating* de um comentário e, posteriormente, eles aplicaram um algoritmo baseado em *metric labing* que, em alguns casos, pode superar o desempenho de algumas versões do SVM e a *baseline* de humanos na classificação de sentimentos em dados com três ou quatro classes.

Goldberg e Zhou [GOLDBERG e ZHU 2006] apresentaram um algoritmo semisupervisionado baseado em grafos a fim de inferir *ratings*, utilizando, em parte, dados não classificados, isto é, não rotulados. Para cada opinião não classificada *x,* esta foi conectada com outras *k* vizinhas previamente classificadas. Além disso, a opinião *x* também foi conectada com suas vizinhas *k'* não rotuladas. Esse grafo criado com tais relações foi utilizado como treinamento para algoritmos de aprendizado, onde a função *f(x)* foi utilizada para suavizar o grafo. Como experimento, eles usaram cinco algoritmos de aprendizado baseados em regressão e em *metric labeling*, demonstrando o benefício em utilizar opiniões não rotuladas no problema de inferência de *rating*.

Tabela 8. Resumo dos principais trabalhos em análise de sentimento binária

Autores	Domínio	Seleção de Características	Algoritmos	Acurácia (%)
Pang *et al.* 2002	Filmes	POS, unigramas, bigramas, posição, adjetivos	NB, EntMax e SVM	82.9 (SVM + unigramas)
Mak et al 2003	Filmes	IG e DF	Decision Tree, kNN e NB	65 (DT + DF)
Matsumoto *et al.* 2005	Filmes	Unigramas, bigramas, frequentes subsequências de palavras e sub-árvores dependentes	SVM	93.7 (SVM + unigramas + bigramas, frequentes subsequências de palavras
Tan e Zhang 2008	Educação, filmes e eletrodomésticos	IG, DF and CHI	Classificador centroide, kNN, NB, Winnow e o SVM	*90.6 (SVM + IG) – Medida Macro F1
Go *et al.* 2009	Tweets	Palavras com sentimento, bigramas and unigramas	NB, EntMax e SVM	83.0 (EntMax com unigramas + bigramas)
Paltoglou e Thelwall 2010	Filmes	Unigramas e DF – variantes do *tfidf*	SVM	96.9 (SVM + BM25 *tf* + *variante* BM25 delta *idf*)[b]
Kang *et al.* 2011	Restaurantes	Unigramas and bigramas	NB, SVM e NB adaptado	81.2 (NB adaptado + unigramas + brigramas)
Xia *et al.*, 2011	Livros, eletrônicos, DVD's e artigos de cozinha	POS and dependência sintática (Word Relation - WR)	NB, SVM e EntMax	Filme – 86.85 (EntMax + POS) Cozinha – 88.65 (NB + WR)
Sharma e Dey 2012	Filmes	IG, GR, MI, CHI e Belief	NB, SVM, EntMax, DT, kNN, Adaboost e Winnow	90.9 (NB + GR)
Ortigosa *et al.* 2014	*Posts* no Facebook	Classificação léxica	J48, NB e SVM	83.27 (SVM + classificador léxico)

Analisando dados do *Twitter*, Pak e Paroubek]PAK e PAROUBEK 2010] coletaram microtextos e os separaram em três classes: sentimento positivo, sentimento negativo e textos objetivos. Esses *tweets* foram selecionados a partir de *emoticons* que apresentassem uma relação com os sentimentos "felizes" ou "tristes". As TEC's utilizadas para o treinamento foram n-gramas e a frequência dos mesmos nos *tweets* selecionados. Entretanto, para o treinamento do classificador utilizado (Naive Bayes), eles utilizaram, além de n-gramas, marcadores POS. Como resultado final, eles demonstram que o melhor resultado foi utilizando bigramas, com acurácia chegando a 85%.

Qu *et al.* [QU; IFRIM; WEIKUM 2010] introduziram um novo tipo de *bag-of-opinions*. Seja uma opinião composta de várias frases, cada frase foi assinalada com um *score* e o *rating* foi inferido agregando os resultados dos *scores*. Para determinar o *score*, um método de regressão foi utilizado, no qual o modelo foi inferido baseando-se nos valores de todas as frases por meio de um modelo de n-gramas proposto. Este modelo avalia o *score* de cada unigrama e, por fim, gera um *score* final para uma frase, no qual um unigrama é o foco da frase (raiz), seguido de n-gramas modificadores e negadores. Os autores mostraram que esta técnica supera todos os trabalhos anteriores em uma margem significante.

Long *et al.* [LONG; ZHANG; ZHUT 2010] propuseram uma nova pesquisa em seleção de opiniões a fim de estimar os *ratings* para serviços em sites utilizando a distância de informação das opiniões por meio da complexidade Kolmogorov. O modelo Kolmogorov associa um valor numérico a cada *string* binária e induz um conceito de similaridade entre tais *strings*. Neste trabalho, a inferência do *rating* foi feita em relação a um atributo do serviço. Isto é, seja um item A com vários atributos $a_1, a_2, \ldots, a_n$. Para inferir o *rating* para A, os autores utilizaram uma combinação dos valores inferidos para cada atributo a por meio de classificadores de redes Bayesianas. Este método produziu bons resultados para o problema de análise de sentimentos multiclasse usando qualquer tipo de opiniões, sejam elas compreensíveis (quando estão relacionadas especificamente sobre os atributos de uma entidade) ou não (quando algum atributo não possui uma

opinião) em relação aos atributos utilizados: preço, serviço, quartos e limpeza. Quando o resultado foi estimado para opiniões compreensíveis, a acurácia, entretanto, não chega a 60%.

Albornoz *et al.*]DE ALBORNOZ *et al*. 2011] analisaram o impacto de diferentes características de um produto e o *rating* final. O objetivo é inferir o *rating* com base no *rating* que cada atributo que um produto recebeu em uma determinada avaliação. Para isso, os autores criaram um vetor com a *intensidade dos atributos*, que foi baseado na polaridade e na força da opinião expressada e em outras opiniões associadas a ela, utilizado para o treinamento dos algoritmos de aprendizado. Em relação aos resultados, o algoritmo Logistic (disponível na ferramenta Weka) apresentou o melhor resultado, atingindo 46,9% de acurácia em relação à 5 classes.

Embora também tenham como principal objetivo a classificação multiclasse, Paltoglou e Thelwall [PALTOGLOU e THELWALL 2013] exploraram outros dois tipos de dimensão afetiva para classificar as opiniões: a valência e a excitação. Eles construíram os vetores de características por meio de *tokens* extraídos considerando as duas dimensões afetivas citadas e utilizam um modelo de regressão e uma variação do algoritmo SVM (OVA) para classificar uma opinião em uma escala de sentimento (escala 1-5).

Na Tabela 9, os trabalhos estão organizados destacando o domínio, as TEC's e os algoritmos empregados e a acurácia final. Como citado por Pang e Lee [PANG e LEE 2008], o problema de multiclasse pode ser resolvido por meio da regressão, já que os *rating* são ordinais. Isso pode justificar a escolha da grande maioria dos autores pela utilização do SVM e outros modelos de regressão.

Tabela 9. Resumo dos principais trabalhos em análise de sentimento multiclasse

Autores	Domínio	Técnicas de Extração de Características	Algoritmos	Acurácia (%)
Pang e Lee 2005	Filmes	Frequência de um termo	SVM One-vs-all, Regression and Metric label	59,4 (SVM + vetor de palavras + regressão)
Goldberg e Zhou 2006	Filmes	Modelo semi-supervisionado baseado em grafos	Regressão (SVM), Metric Labeling e PSP	59,2 (regressão+PSP ou regressão)
Pak e Paroubek	*Tweets*	Frequência, n-gramas e POS	NB	60-80 (NB + bigramas)
Qu *et al.*2010	Produtos da Amazon	Bag of opinions	Regressão	-*mostra apenas o erro quadrático médio
Long *et al.* 2010	Hotéis	Complexidade Kolmogorov + rede bayesiana	SVM	73,1 – 57,3 (Kolmogorov+SVM baseado em atributos)
Albornoz *et al.*, 2011	Hotéis	Vetor de intensidade das características	Regressão logística, SVM e Árvore funcional	46,9 (vetor+regressão)
Paltoglou e Thelwall 2013	Notícias	Palavras que expressem valência e excitação	Regressão de Vetor de Suporte, SVM (OvA)	51,8 (Excitação + SVM (OvA))

1.7.4 Aplicações da Análise de Sentimentos

Além das pesquisas voltadas para a comparação entre técnicas de aprendizado e seleção de características, pode-se encontrar trabalhos que, a partir do uso das mesmas, apresentam também uma aplicação final. Nos exemplos abaixo, destaque para trabalhos voltados para as áreas de educação e serviços.

Chen *et al.* [CHEN *et al*. 2006] criam uma análise visual de opiniões positivas e negativas do livro "The Da Vinci Code". Eles utilizam uma ferramenta visual, o TermWatch, para construir uma rede multicamada de termos baseada em associações sintáticas, semânticas e estatísticas. A fim de avaliar os termos que foram selecionados

anteriormente, eles utilizam um modelo preditivo baseado no SVM. Como característica para o treinamento, um conjunto de opiniões positivas e negativas é utilizado. Neste caso, uma opinião é decomposta em três componentes que refletem a presença de termos positivos, negativos e comuns em ambas as categorias.

Mak *et al.* [MAK *et al*. 2003] criaram um sistema de recomendação web utilizando categorização textual de sinopses de filmes armazenadas no IMDB, selecionados do EachMovie database. Primeiramente, eles adaptaram as opiniões a fim de serem utilizadas nos algoritmos, representando-as em vetores, retirando palavras que não possuem informação útil, utilizando valores para cada palavra restante e ranqueando as características do corpus resultante através de três TECs: IG, DF e Informação Mútua. Com essa primeira etapa finalizada, eles utilizaram três algoritmos para construir um classificador para um usuário do sistema: kNN, Decisions Trees e o Naive Bayes. O desempenho final dos algoritmos foi em torno de 60 a 65%, com as árvores de decisão apresentando o melhor resultado, entretanto, a diferença entre os três é pouco significativa.

Em [GO *et al*. 2009], Go *et al.* utilizaram *emoticons* a fim de treinarem opiniões retiradas do Twitter, utilizando algoritmos de aprendizado. Além dos *emoticons*, palavras-chave presentes no site Twittratr[16] que tenham sentimento positivo ou negativo foram utilizadas no treinamento como unigramas e bigramas. Após testes, eles atingiram cerca de 83% de acurácia com o algoritmo Naive Bayes configurado tanto com unigramas como bigramas. Por fim, os autores também disponibilizaram um site, o *sentiment140*[17], onde é possível saber o sentimento sobre algo em relação aos tweets existentes. O site cria uma lista de tweets positivos, negativos e neutros, além de gráficos que mostram qual sentimento é predominante.

Na área de educação, Ortigosa *et al.* [ORTIGOSA *et al*. 2014] construíram um modelo para avaliar postagens no *Facebook*™[18] e, a partir da detecção do sentimento habitual do usuário, verificar mudanças emocionais. A aplicação é chamada de SentBuk. Essa informação foi utilizada em sistemas e-learning a fim de recomendar atividades mais adequadas em relação ao humor do estudante em determinado período. Eles construíram um classificador léxico e, quando um grande número de *posts* foi classificado, eles usaram essas mensagens como entrada de treinamento para o algoritmo de aprendizado de máquina. Para realizar os testes eles utilizaram os algoritmos J48, Naive-Bayes e SVM (radial e sigmoide), onde o melhor resultado foi utilizando o algoritmo SVM (sigmoide) com 83% de acurácia.

1.7.5 Dificuldades da Análise de Sentimentos

A grande maioria dos trabalhos de mineração de opiniões existentes tem como foco a mineração de opiniões no idioma inglês. Embora raros, trabalhos como o de Ortigosa *et al.* [ORTIGOSA; MARTÍN; CARRO 2014] e Tang e Zhang [TAN e ZHANG 2008] utilizam os idiomas espanhol e mandarim, respectivamente. Essa falta de trabalhos em alguns idiomas pode dificultar a análise já que os idiomas têm processos de construção diferentes.

[16] http://twitrratr.com/

[17] http://www.sentiment140.com/

[18] http://facebook.com

Em relação ao pré-processamento textual, nota-se uma dificuldade na escolha de palavras para treinamento. Isso porque a forma de escrever pode mudar para cada pessoa. Uma expressão que possa indicar um sentimento muito bom para uma pessoa, pode indicar um sentimento nem sempre bom para outra [LIU 2012]. Da mesma forma, uma palavra pode ser utilizada em qualquer classe, logo, o contexto deve ser analisado para compreender o sentimento da mesma. Um exemplo disso são as ironias, muito presentes em avaliações políticas, por exemplo.

Além disso, outra dificuldade está na análise de opiniões que apresentam poucas palavras ou expressões que indiquem algum sentimento. Esse é um caso estudado principalmente em *tweets*. Isso se deve ao fato de um *tweet* ter o número de caracteres limitado a 140. Em alguns casos, como feito em [GO; BHAYANI; HUANG 2009], a utilização de *emoticons* no treinamento é uma opção a fim de melhorar o desempenho da mineração de opiniões. Outro problema está na existência de *herding effects* [WANG e WANG 2014] para a RIP. Isso se deve ao fato de muitas vezes o *rating* final de um usuário não condizer com o comentário. Isso pode acontecer, por exemplo, pelo fato do *rating* ser baseado na média de notas existente no site e não avaliado em relação à opinião por si só. Muitas vezes, pode-se notar que um comentário possui uma avaliação que poderia ter nota máxima (5 estrelas) mas tem o *rating* 4, por exemplo.

1.8 Conclusão

Embora existam muitas técnicas de análise de sentimentos, como foi notado na discussão dos trabalhos e pesquisas da Seção 1.7, as principais técnicas de extração de características foram apresentadas, bem como alguns dos principais algoritmos de aprendizado utilizados na mineração de opiniões que foram selecionados de acordo com a importância e relevância dos trabalhos relacionados.

Esses tipos de classificação (binária e multiclasse) estão amplamente presentes em sistemas e-commerce e são fundamentais para os usuários desse sistema, capturar e compreender de forma correta o sentimento de outros usuários em relação a um item. Por meio dessas técnicas apresentadas, um bom sistema de análise de opiniões pode ser criado para qualquer tipo de domínio.

O problema de classificação binária, modelos de aprendizado podem ser empregado em sistemas semelhantes ao *sentiment140*. Para problemas multiclasse, esse tipo de análise de sentimentos pode ser usado tanto para criar resumos de sentimentos, processo conhecido como sumarização, ou para sistemas de recomendação assistida de *ratings*, no qual estrelas podem ser inferidas para a opinião dos usuários.

Um estudo de caso baseado em avaliações retiradas do site TripAdvisor™[19], disponíveis em[20], que foi utilizado na pesquisa de [WANG *et al*. 2010], foi realizado com as técnicas e algoritmos descritos abaixo e pode ser encontrado no link[21] abaixo.

Referências

Aly, M. (2005) “Survey on Multiclass Classification Methods Extensible Algorithms. Neural Networks”, N. November, P. 1–9.

[19] http://tripadvisor.com
[20] http://times.cs.uiuc.edu/~wang296/Data/
[21] http://www2.ic.uff.br/PosGraduacao/Dissertacoes/722.pdf

Baccianella, S., Esuli, A. and Sebastiani, F. (2010) “Sentiwordnet 3.0: an Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining”, Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10), V. 0, N. November, P. 2200–2204.

Beineke, P., Hastie, T., Manning, C. and Vaithyanathan, S. (2004) “Exploring Sentiment Summarization”, Proceedings Of The AAAI Spring Symposium On Exploring Attitude And Affect In Text Theories And Applications, V. 07, P. 1–4.

Brooke, J. (2009) “A Semantic Approach to Automated Text Sentiment Analysis”, Simon Fraser University, V. 26, N. 4, P. 118.

Cambria, E., Schuller, B., Xia, Y.and Havasi, C. (2013) “New Avenues in Opinion Mining and Sentiment Analysis. IEEE Intelligent Systems, N. April, P. 15–21.

Chen, C., Ibekwe-Sanjuan, F., Sanjuan, E. and Weaver, C. (2006) “Visual Analysis of Conflicting Opinions”, IEEE Symposium on Visual Analytics Science and Technology 2006, Vast 2006 - Proceedings, P. 59–66.

Das, S. R and Chen, M. Y. (2001) “Yahoo! For Amazon: Opinion Extraction From Small Talk on the Web. Proceedings of the 8th Asia Pacific Finance Association Annual Conference, V. Xxxiii, N. 2, P. 81–87.

Dave, K., Lawrence, S. and Pennock, D. (2003) “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews”, Proceedings of the 12th International Conference on World Wide Web, P. 519–528.

De Albornoz, J. C., Plaza, L., Gervás, P. and Díaz, A. (2011) “A Joint Model of Feature Mining and Sentiment Analysis for Product Review Rating”, Advances in Information Retrieval, p. 55–66.

Dumais, S., Platt, J., Heckerman, D. and Sahami, M. (1998) “Inductive Learning Algorithms and Representations for Text Categorization”, Cikm ’98: Proceedings of the Seventh International Conference on Information and Knowledge Management, P. 148–155.

Go, A., Bhayani, R. and Huang, L. (2009) “Twitter Sentiment Classification Using Distant Supervision”, Processing Cs224n Project Report, Stanford, V. 150, N. 12, P. 1–6.

Godbole, S. and Sarawagi, S. (2004) “Discriminative Methods for Multi-Labeled Classification”, Advances in Knowledge Discovery and Data, V. Lncs3056, p. 22–30.

Goldberg, A. B. and Zhu, X. (2012) “Seeing Stars When There aren’t Many Stars: Graph-Based Semi-Supervised Learning for Sentiment Categorization”, Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing.

Harrington, P. (2012) “Machine Learning In Action”, Manning, 2012.

Kaestner, C. A. A. (2013) “Support Vector Machines and Kernel Functions for Text Processing”, Revista de Informática Teórica E Aplicada, P. 1–7.

Kang, H., Yoo, S. J. and Han, D. (2012) “Senti-Lexicon and Improved Naïve Bayes Algorithms for Sentiment Analysis of Restaurant Reviews”, Expert Systems with Applications, V. 39, N. 5, p. 6000–6010.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R. and Riedl, J.

(1997) “Grouplens: Applying Collaborative Filtering to Usenet News”, Communications of the Acm, V. 40, N. 3, P. 73–75.

Kontopoulos, E., Berberidis, C. and Dergiades, T. (2013) “Ontology-Based Sentiment Analysis of Twitter Posts”, Expert Systems with Applications, V. 40, N. 10, p. 4065–4074.

Likert, R. (1932) “A Technique for the Measturement of Attittudes”, Archives of Psychology, V. 22, N. 140, P. 1–55.

Liu, B. (2012) “Sentiment Analysis and Opinion Mining” Morgan and Claypool Publishers, N. May.

Long, C., Zhang, J. and Zhut, X. (2010) “A Review Selection Approach for Accurate Feature Rating Estimation”, Proceedings of the 23rd International Conference on Computational Linguistics: Posters, N. August, p. 766–774.

Lunardi, A. C., Viterbo, J. and Bernardini, F. C. (2015) “Um Levantamento do Uso de Algoritmos de Aprendizado Supervisionado em Mineração de Opiniões”, ENIAC - Natal, RN.

Mak, H., Koprinska, I. and Poon, J. (2003) “Intimate: A Web-Based Movie Recommender Using Text Categorization”, Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003), p. 2–5.

Martineau, J. and Finin, T. (2009) “Delta *tfidf* : An Improved Feature Space for Sentiment Analysis”, ICWSM, May, p. 258–261.

Matsumoto, S., Takamura, H. and Okumura, M. (2005) “Sentiment Classification Using Word Sub-Sequences and Dependency Sub-Trees”, Proceedings of the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, V. 05 the 9, p. 301–311.

McCallum, A. and Nigam, K. (1998) “A Comparison of Event Models for Naive Bayes Text Classification”, AAAI/ICML-98 Workshop on Learning for Text Categorization, p. 41–48.

Mitchell, T. M. (1997) “Machine Learning”..

Nasukawa, T. and Yi, J. (2003) “Sentiment Analysis : Capturing Favorability Using Natural Language Processing Definition of Sentiment Expressions”, 2nd International Conference on Knowledge Capture, p. 70–77.

Ortigosa, A., Martín, J. M. and Carro, R. M. (2014) “Sentiment Analysis in Facebook and its Application to e-Learning”, Computers in Human Behavior, v. 31, p. 527–541.

Pak, A. and Paroubek, P. (2010) “Twitter as a Corpus for Sentiment Analysis and Opinion Mining”, LREC, p. 1320–1326.

Paltoglou, G. and Thelwall, M. (2012) “A Study of Information Retrieval Weighting Schemes for Sentiment Analysis”, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, n. July, p. 1386–1395.

Paltoglou, G. and Thelwall, M. (2013) “Seeing Stars of Valence and Arousal in Blog Posts”, IEEE Transactions on Affective Computing, v. 4, n. 1, p. 116–123.

Pang, B. and Lee, L. (2005) “Seeing Stars: Exploiting Class Relationships for Sentiment

Categorization with Respect to Rating Scales", In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (p. 115-124). Association For Computational Linguistics. v. 3, v. 1.

Pang, B. and Lee, L. (2008) "Opinion Mining And Sentiment Analysis", Foundations and Trends in Information Retrieval, v. 2, n. 1, p. 1–135.

Pang, B., Lee, L. and Vaithyanathan, S. (2002) "Thumbs Up? Sentiment Classification Using Machine Learning Techniques", Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10, n. July, p. 79–86.

Pimenta, E. (2004) "Abordagens para a Decomposiçao de Problemas Multiclasse: os Códigos de Correção de Erro e Saída", Dissertação da Universidade do Porto, 2004.

Platt, J. C. (1998) "Fast Training of Support Vector Machines Using Sequential Minimal Optimization", Advances in Kernel Methods, p. 185 – 208.

Prusa, J. D., Khoshgoftaar, T. M. and Dittman, D. J. (2015) "Impact of Feature Selection Techniques for Tweet Sentiment Classification", The Twenty-Eighth International FLAIRS Conference, p. 299–304.

Qu, L.; Ifrim, G.; Weikum, G. The Bag-Of-Opinions Method For Review Rating Prediction From Sparse Text Patterns. Coling, N. August, P. 913–921, 2010.

Quinlan, J. R. (1986) "Induction of Decision Trees", Machine Learning, v. 1, n. 1, p. 81–106.

Sharma, A. and Dey, S. (2012) "A Comparative Study of Feature Selection and Machine Learning Techniques for Sentiment Analysis", RAC'S 2012, p. 1–7.

Tan, S. and Zhang, J. (2008) "An Empirical Study of Sentiment Analysis for Chinese Documents", Expert Systems with Applications, v. 34, n. 4, p. 2622–2629.

Tang, H., Tan, S. and Cheng, X. (2009) "A Survey on Sentiment Detection of Reviews", Expert Systems with Applications, v. 36, n. 7, p. 10760–10773.

Turney, P. D. (2002) "Thumbs Up Or Thumbs Down ? Semantic Orientation Applied to Unsupervised Classification of Reviews", Proceedings of the 40th Annual Meeting on Association for Computational Linguistics.

Wang, H., Lu, Y. and Zhai, C. (2010) "Latent Aspect Rating Analysis on Review Text Data", Proceedings of the 16th ACM SigKDD International Conference on Knowledge Discovery and Data Mining - KDD'10, p. 783.

Wang, T. and Wang, D. (2014) "Why Amazon's Ratings Might Mislead You: The Story of Herding Effects", Big Data, v. 2, n. 4, p. 196–204.

Wiebe, J. M. and Rillof, E. (2005) "Creating Subjective and Objective Sentence Classifiers from Unannotated Texts", Computational Linguistics and Intelligent Text Processing, v. 3406, p. 486–497.

Xia, R., Zong, C. and Li, S. (2011) "Ensemble of Feature Sets and Classification Algorithms for Sentiment Classification", Information Sciences, v. 181, n. 6, p. 1138.