



Um estudo comparativo sobre métodos de análise de sentimentos em *tweets*

Alison Pereira Ribeiro, Nádia F. F. da Silva
Instituto de Informática – Universidade Federal de Goiás
alisonrib17@gmail.com, nadia@inf.ufg.br

Resumo—O Twitter é um *microblog* em que os usuários podem postar atualizações (*tweets*) para amigos (seguidores). A *Análise de Sentimentos* tem se tornado um importante campo de estudo neste ambiente devido à enorme quantidade de *tweets* disponíveis, o que possibilita diversas aplicações como monitoramento de marcas e produtos, previsão de campanhas políticas e até aplicações no mercado financeiro. Um dos grandes desafios da análise de sentimentos em *tweets* está na criação de modelos preditivos que são capazes de classificar-los como positivo, negativo ou neutro. Os principais modelos propostos na literatura utilizam de abordagens baseadas em processamento de linguagem natural e aprendizado de máquina. Frente o contexto apresentado, este artigo visa comparar o desempenho dos seguintes métodos de análise de sentimentos: aprendizado de máquina, dicionários léxicos, *emoticons*, *part-of-speech*, *ensembles* e *word embeddings*. O objetivo é indicar ao leitor, dentre tais abordagens, a que melhor se adéqua às particularidades dos *tweets*. Os experimentos foram aplicados em duas bases de dados, Sanders e HCR. Em ambos conjuntos de dados, os procedimentos que obtiveram os melhores resultados foram baseados em dicionário léxico e *word embeddings* com 79,09% e 79,36% de acurácia, respectivamente, para Sanders. Enquanto que para HCR o resultado foi 69,11% e 68,22% de acurácia, respectivamente.

Palavras-chave—Análise de Sentimentos, Aprendizado de Máquina, Dicionários léxicos, Part-of-Speech, Ensembles, Word Embeddings.

A comparative study on methods of sentiment analysis in tweets

Abstract—Twitter is a microblog on which users can post updates (*tweets*) to friends (followers). Sentiment Analysis has become an important field of study in this environment due to the sheer number of tweets available, which allows several applications such as monitoring of brands and products, forecasting political campaigns and even applications in the financial market. One of the great challenges in analyzing feelings is in the creation of predictive models that are able to classify tweets as positive, negative or neutral. The main models are based on natural language processing and machine learning. Given this context, this article aims to compare the performance of the following methods of sentiment analysis: machine learning, lexical dictionaries, emoticons, part-of-speech, ensembles and word embeddings. The objective is to indicate to the reader, among such approaches, which best suits

the particularities of the tweets. The experiments were applied to two databases, Sanders and HCR. In both datasets, the procedures that obtained the best results were based on lexical dictionary and word embeddings with 79.09% and 79.36% of accuracy, respectively, for Sanders. While for HCR the result was 69.11% and 68.22% accuracy, respectively.

Index Terms—Sentiment Analysis, Machine Learning, Lexical dictionaries, Part-of-Speech, Ensembles, Word Embeddings.

I. INTRODUÇÃO

DE acordo com [1], descobrir o que as pessoas pensam sempre foi motivo de interesse, e plataformas como o Twitter¹ e o Facebook², possibilitam que seus usuários possam expressar opiniões sobre algum evento específico – podendo ser esportivo, político ou até opiniões direcionadas a empresas e serviços. Devido ao crescimento de ambientes que dispõem de uma grande quantidade de dados subjetivos, a tarefa de classificar sentimentos passou a ser objeto de estudo [2]–[9].

O Twitter é uma fonte importante para realização de pesquisas por ser uma plataforma que propicia a difusão de conteúdo. Por meio dos chamados *tweets*, os usuários expõem o que pensam dentro de 280 caracteres³. Os *tweets* apresentam algumas particularidades como abreviações, gírias e múltiplos contextos⁴ [10]. Por causa dessas características ocorre uma esparsidade nos dados, e isso gera um impacto sobre o desempenho global da análise de sentimento. Outra razão para a esparsidade nos dados é o fato de que uma grande porcentagem dos termos que aparecem nos *tweets* ocorrem menos de 10 vezes [11].

Além dos problemas citados anteriormente, o tratamento da negação [12], a construção de listas de *stop words*⁵ para o Twitter [13], a variação de tópicos, o contexto multilíngue [14] e a *tokenização* [14] são problemas

¹<https://twitter.com/>

²<https://www.facebook.com>

³<https://techcrunch.com/2017/11/07/twitter-officially-expands-its-character-count-to-280-starting-today/>

⁴ *Weblogs*, comentários em sites de vendas de produtos específicos e fóruns, ao contrário do Twitter, apresentam propensão a estabelecer contextos únicos.

⁵ *Stop Words* são palavras que têm baixo poder de discriminação (por exemplo, “a”, “é”, “que” etc.) que tendem a ser filtradas antes de se processar o texto.

que impactam diretamente o resultado da tarefa de análise de sentimentos em *tweets*, e precisam ser tratados com rigor.

Frente aos desafios, pesquisadores utilizam diversos métodos preditivos presentes na literatura a fim de minimizar os problemas. Como não há um consenso estabelecido sobre qual a melhor abordagem para a classificação de sentimentos, é livre para o pesquisador escolher entre os métodos preditivos: (i) baseados em aprendizado de máquina, (ii) e/ou baseados em dicionários léxicos⁶, (iii) e/ou métodos que incluem características baseadas em orientação sintática (aspectos gramaticais), (iv) e/ou métodos que utilizam artifícios de escrita característicos da rede social, como os *emoticons*⁷ e as *hashtags* [15].

É importante ressaltar que em todos os casos a engenharia de atributos ou tarefa de definição das características que melhor representam os textos se faz necessária, por exemplo: unigrama, bigrama, etc.

O presente trabalho tem como objetivo explorar e comparar o desempenho de diversos modelos preditivos quando expostos às principais técnicas de engenharia de atributos utilizadas na literatura. Particularmente, a ideia principal é prover para análise de sentimentos um estudo focado em métodos baseados em dicionários léxicos, aprendizado de máquina, orientação sintática, *emoticons*, *ensembles*⁸ e *word embeddings*. E também levantamos algumas questões a serem respondidas: (i) Qual método possui a melhor performance? (ii) Qual modelo de *ensemble* é capaz de obter o melhor resultado? (iii) Quantas palavras em comum existem nos léxicos utilizados nos experimentos? (iv) E quantas dessas palavras se contradizem nos léxicos escolhidos? (v) Qual é o resultado ao utilizar uma representação vetorial de palavras (*word embeddings*)?

O artigo está organizado da seguinte forma: a Seção 2 discorre sobre os trabalhos relacionados, a Seção 3 explica alguns conceitos relacionados à Análise de Sentimentos, a Seção 4 apresenta uma descrição das bases de dados, a Seção 5 aborda a metodologia do trabalho e como cada método foi aplicado, a Seção 6 mostra os resultados obtidos. Por fim, a Seção 7 apresenta as considerações finais e uma descrição dos trabalhos futuros.

II. REVISÃO BIBLIOGRÁFICA

A quantidade de trabalhos na área de Análise de Sentimentos vem crescendo a cada ano, com muitos pesquisadores fazendo esforços para combinar conceitos de aprendizado de máquina e processamento de linguagem natural nos últimos anos. Esta seção descreve brevemente alguns dos inúmeros estudos relacionados à análise do sentimento em *tweets*.

⁶O dicionário léxico é um conjunto de palavras armazenadas em um *dataset*, construído automaticamente ou manualmente, que polariza as palavras que o compõe de acordo com os sentimentos daquele contexto.

⁷Os *emoticons* são 'símbolos' gráficos que agregam indícios de emoções e sentimentos.

⁸Também conhecido na literatura como Agregadores de classificadores ou Comitês de classificadores. O leitor interessado em maiores detalhes sobre o assunto pode consultar [16].

Em [17], os autores apresentaram um trabalho no intuito de comparar 8 métodos baseados em léxicos de sentimentos. A partir dos 8 métodos, 7 foram escolhidos para um novo experimento chamado pelos autores de Método Combinado. Na análise feita em cada método, a comparação foi realizada através da acurácia e *F-measure*. Os métodos que obtiveram melhores resultados foram o SenticNet [6] e o SentiWordNet [18]. Já no Método Combinado, os autores realizaram a seguinte estratégia: analisaram a média harmônica (*F-measure*), a precisão e *recall* de cada método e distribuíram diferentes pesos para cada um deles. A combinação foi feita de forma incremental e o resultado foi de *F-measure* igual a 0,730%.

Enquanto que em [19], foi realizado um estudo destinado à análise de sentimentos em 9 idiomas. Para isso, os autores selecionaram 13 métodos baseados em aprendizado de máquina e dicionários léxicos. Nesse trabalho, as bases de dados foram traduzidas para o idioma inglês. Os autores tiveram como objetivo verificar a qualidade dos métodos abordados. Para isso, cada método foi comparado em nível de abrangência, i.e, a porcentagem de mensagens na base em que cada método conseguiu detectar algum sentimento (positivo ou negativo). Os métodos (dicionários léxicos) que se destacaram foram: SentiWordNet [18], Sentiment140 *Lexicon* [20] e SenticNet [6].

O sistema proposto por [21], construiu um modelo de aprendizado de máquina (*machine learning*) para detectar *tweets* positivos e negativos. Esse modelo utilizou diferentes técnicas para representar os tweets de entrada marcados na fase de treinamento, usando diferentes *features sets*, tais como: *bag-of-words*, *lexicon-based*, *PoS features* e *emoticon features*. Os autores aplicaram um método *ensemble* baseado no SVM, NB e LR. Essa configuração foi capaz de obter 93,94% e 84,75% de acurácia, nas bases de dados Sanders e HCR, respectivamente.

Nos estudos de Chaovalit e Zhou [22], duas abordagens foram usadas: aprendizado de máquina e *part-of-speech*⁹. A abordagem de aprendizado de máquina foi empregada com *n-gramas*¹⁰ em duas formas, a primeira aplicando validação cruzada com 3 *folds* e a segunda com uma base de dados para teste. A acurácia obtida foi de 84,49% e 66,27%, respectivamente. Já a abordagem de *part-of-speech* alcançou uma acurácia de 77%. O desempenho da técnica de aprendizado de máquina foi influenciado pelo uso de *n-gramas* juntamente com a eliminação de ruídos. Enquanto o método de *part-of-speech*, teve a performance diretamente ligada ao conjunto de *tags* aplicadas na análise. Os testes realizados foram feitos em avaliações de filmes porque segundo os autores é mais desafiador devido as palavras irônicas presentes nas críticas.

Em [5], os pesquisadores propuseram uma estrutura para análise de sentimentos usando uma abordagem baseada em dicionário léxico e trazendo um estudo comparativo sobre técnicas de mineração de opinião, incluindo

⁹*Part-of-speech* é um processo de rotulação de elementos textuais – tipicamente palavras e pontuação – com o objetivo de evidenciar a estrutura gramatical de um determinado trecho de texto [23].

¹⁰Termos compostos por *n* palavras.

aprendizado de máquina e léxicos. As comparações foram embasadas em recursos como técnica empregada, dicionários e abordagens de *soft-computing* [24]. Os autores propuseram também uma abordagem para dicionário léxico que incorpora uma lógica difusa, este método trata, principalmente, de classificar as avaliações como positivas, negativas ou neutras com base em uma pontuação que é calculada usando os dicionários SentiWordNet [20] e WordNet [25].

Diante do desafio do *SemEval-2014 Task 9: Sentiment Analysis in Twitter* [26], foi proposto por [27] uma abordagem híbrida para classificação de sentimentos no *Twitter*. O sistema combina técnicas como *machine learning* e *lexicon-based*¹¹. Os autores utilizaram SVM para classificar as mensagens e o modelo híbrido teve o melhor resultado com *f-measure* igual a 63,94.

Mansar, Gatti, Ferradans et al. [8], propuseram uma abordagem léxica-afetiva e uma representação vetorial de palavras combinados com redes neurais convolucionais para inferir o sentimento das manchetes de notícias financeiras. Essa arquitetura foi utilizada e avaliada no contexto do desafio SemEval 2017 (tarefa 5, subtarefa 2), na qual obtiveram o primeiro lugar, com 0,745 de *cosine-similarity*¹². Como *word embedding* os autores utilizaram o modelo GloVe pré-treinado (treinado no wikipédia+gigaword corpus) e o léxico *DepecheMood* [29].

Para o mesmo problema, Rotim, Tutek e Šnajder [9] aplicaram modelos pré-treinados do GloVe e *Skip-gram* (*Word2Vec*) Google News corpus¹³ com 300 dimensões. Este último rendeu aos autores o terceiro lugar na competição, alcançando 0,733 de *cosine-similarity*.

Enquanto que [30] propôs para o SemEval-2017 uma abordagem híbrida, baseada em *bag-of-words*, *word embeddings* (*Word2Vec*) e *sentiment lexicon*. Essa abordagem mostrou-se eficaz, uma vez que obteve 0,6417 de *cosine-similarity*.

Com relação aos trabalhos relacionados, este trabalho se difere ao realizar um estudo comparativo entre 7 métodos de análise de sentimentos. Este estudo é baseado em experimentos no qual os métodos são explorados de modo independente e também combinados entre si, assim gerando novas configurações ausentes nos trabalhos relatados.

De forma a facilitar a compreensão, o leitor pode conferir na Tabela I um resumo sobre os trabalhos relacionados.

III. ANÁLISE DE SENTIMENTOS

A Análise de Sentimentos é o campo da Ciência da Computação responsável por realizar pesquisas de mineração de dados, linguística computacional, recuperação de informações, inteligência artificial, entre outras. A análise de sentimentos pode ser definida como qualquer

estudo feito computacionalmente envolvendo opiniões, sentimentos, avaliações, atitudes, afeições, visões, emoções e subjetividade, expressos de forma textual [4]. Trata-se de uma área em expansão que engloba diversas técnicas, já citadas na seção I. A análise de sentimentos consiste em 4 fases: seleção de dados relevantes, pré-processamento, classificação e predição.

Em AS (Análise de Sentimentos), nota-se um predomínio do uso de métodos supervisionados [31]–[35], mais especificamente, classificação e regressão. O problema da classificação pode ser dividido em dois passos: (i) aprender um modelo de classificação sobre um *corpus* de treinamento previamente rotulado como positivo, negativo ou neutro; (ii) prever a polaridade¹⁴ de novos textos com base no modelo resultante (modelo anteriormente treinado).

Os dados para a classificação correspondem a um conjunto de instâncias caracterizadas por atributos. O rótulo é denominado atributo-alvo, enquanto que os demais são designados como atributos discriminantes ou *features* [36]. Em termos de pré-processamento, é necessário extrair de cada porção de texto analisado, as *features* relevantes para a tarefa de classificação e representá-las na forma de um vetor de termos, chamado de *bag-of-words*. A Figura 1 ilustra uma coleção de *tweets* após a etapa de pré-processamento.

	t1	t2	...	tn
tweet1	a11	a12	...	a1m
tweet2	a21	a22	...	a2m
...
tweetn	an1	an2	...	anm

Fig. 1. Representação de uma *bag-of-words*. As linhas representam cada um dos *tweets*, e as colunas representam as *features* relevantes extraídas dos *tweets*.

Isso significa que o valor a_{ij} refere-se ao valor associado ao j -ésimo termo do *tweet* i , isto é, a_{ij} é o valor do termo t_j no *tweet* i e pode ser calculado pela frequência ou pela presença. Alguns autores utilizam valores binários [10], [15]. Nesse caso, $a_{ij} = 0$ significa a ausência do termo j na mensagem i e valores maiores que 0 (zero) significam a presença ou quantidade do termo. Após esta etapa, o algoritmo de classificação recebe como entrada o modelo *bag-of-words*, assim como as classes de cada *tweet*, e apresenta como saída a predição das classes.

Vejamos um exemplo com as seguintes sentenças: *Gostei do celular, mesmo sendo caro* (*tweet1*); *Não gostei desse celular, estraga fácil* (*tweet2*). A Figura 2 apresenta como fica a representação de uma *bag-of-words*.

¹¹A abordagem *lexicon-based* é a integração da terminologia sentimental conhecida e pré-compilada, é principalmente uma abordagem baseada em dicionário classificado e uma abordagem baseada em corpus. Esta abordagem baseia-se no léxico de opinião obtido, que são utilizados na análise do texto [28].

¹²Métrica utilizada na competição.

¹³<https://code.google.com/archive/p/word2vec/>

¹⁴O termo polaridade da opinião refere-se à atribuição do sentimento, percepção ou atitude do público em relação ao alvo da opinião. Por exemplo, refere-se ao ato de atribuir a uma unidade textual (sentença, parágrafo, documento) um sentimento positivo ou negativo.

TABELA I
ESTUDOS EM ANÁLISE DE SENTIMENTOS EM TWEETS.

Estudos	Abordagem	Classificador	Dados
Araújo et al. [17]	Dicionários Léxicos	-	-
Fouad et al. [21]	<i>machine learning, bag-of-words, lexicon-based, PoS features.</i>	Ensemble	Sanders, HCR e Stanford
Chaovalit e Zhou [22]	aprendizado de máquina e <i>part-of-speech</i>	-	Reviews
Rosenthal et al. [26]	<i>machine learning e lexicon-based</i>	SVM	SemEval-2014
Mansar et al. [8]	<i>word embeddings</i> e dicionário léxico	CNN	SemEval-2017
Rotim et al. [9]	<i>bag-of-word, word embeddings (GloVe) e Skip-gram (Word2Vec)</i>	SVR	SemEval-2017
Hardeniya et al. [5]	dicionário léxico e <i>machine learning</i>	-	-
Reis et al. [19]	dicionário léxico e <i>machine learning</i>	-	-
Saleiro et al. [30]	<i>bag-of-words e word embeddings e sentiment lexicon</i>	RF, SVR e MLP	SemEval-2017

	caro	celular	desse	estraga	fácil	gostei	mesmo	não	sendo
<i>tweet1</i>	1	1	0	0	0	1	1	0	1
<i>tweet2</i>	0	1	1	1	1	1	0	1	0

Fig. 2. Uma representação de *bag-of-words* baseado nas sentenças citadas. As *features* são as colunas e o alvo das sentenças é o **celular**.

Um dos problemas causados com o uso da estratégia baseada em *bag-of-words* é a esparsidade da tabela resultante devido à grande variabilidade do vocabulário nesses ambientes. Além disso, a definição de contexto também é dificultada, pois a *bag-of-words* não é uma representação que considera a ordem entre as palavras.

Os *word embeddings* [37], modelos de linguagem treinados usando redes neurais profundas, são alternativas para os problemas característicos da abordagem *bag-of-words*. O objetivo desses modelos é prever a palavra seguinte, dado o contexto anterior na frase. Dessa forma, palavras similares tendem a estar sempre próximas no espaço vetorial. A representação vetorial de palavras mostrou-se um grande avanço em relação às estratégias baseadas em *bag-of-words*. O modelo de *word embedding* mais famoso é o *Word2Vec* proposto por Mikolov, Chen, Corrado et al. [7], o *Word2Vec* é um modelo preditivo construído em duas arquiteturas: CBOW e *Skip-gram*. O modelo CBOW (*Continuous Bag of Words*) é treinado para prever uma palavra-alvo a partir de um contexto. Nessa arquitetura, a ordem das palavras não é considerada – razão da referência à representação BOW. O contexto é definido por palavras que tipicamente ocorrem ao redor de uma palavra-alvo (antes e depois), não considerando sua ordem.

A arquitetura *Skip-gram* é semelhante à CBOW, porém o modelo é treinado no sentido inverso: em vez de prever a palavra-alvo, o modelo deve prever as palavras que formam o contexto, a partir de uma palavra central dada como entrada.

Outro modelo que vem sendo explorado na literatura é o GloVe¹⁵ (*Global Vectors*), proposto por Pennington,

Socher e Manning [38], ao contrário do *Word2Vec*, o GloVe é baseado em um modelo de contagem, no qual os vetores são derivados de uma matriz de co-ocorrências usadas para extrair informações estatísticas sobre o corpus. O modelo GloVe é capaz de capturar informação de ordem entre as palavras, o que justifica seu grande uso em análise de sentimentos [8], [9], [39], [40].

IV. DADOS UTILIZADOS

ESTA seção descreve os conjuntos de dados utilizados: *Sanders* [41] e *HCR* [2]. Essas bases de dados foram escolhidas por serem comumente usadas para pesquisas em análise de sentimentos como em [2], [3], [10] e, também, por estarem disponíveis publicamente.

A. Sanders

A base de dados *Sanders* consiste em 5.513 *tweets* classificados manualmente por especialistas, este conjunto de *tweets* foi coletado a partir de quatro tópicos: *@apple*, *#google*, *#microsoft* e *#twitter*. Cada *tweet* possui um rótulo de sentimento: positivo, negativo, neutro e irrelevante. Para este trabalho, apenas os *tweets* classificados como positivo, negativo e neutro foram analisados. Portanto, 3.726 *tweets* foram selecionados, sendo: 570 com sentimento positivo, 653 com sentimento negativo e 2.503 com sentimento neutro.

B. Health Care Reform (HCR)

A base de dados *HCR* foi construída a partir da *hashtag* *#hcr*. Os *tweets* foram coletados em março de 2010 [2]. Esta base de dados foi catalogada em 4 sentimentos: positivo, negativo, neutro e irrelevante. O conjunto de *tweets* foi dividido em dados de treinamento, desenvolvimento e teste, no entanto, mesclamos os dados de treino com os dados de desenvolvimento. Portanto, utilizamos 852 *tweets* para treino e 480 *tweets* para teste. Neste artigo desconsideramos *tweets* dados como irrelevantes.

¹⁵<https://nlp.stanford.edu/projects/glove/>

V. METODOLOGIA

TRATA-SE de um estudo exploratório, que buscou fundamentar-se na literatura com o intuito de criar novas configurações a partir do modelo *bag-of-words* e dos métodos que serão descritos a seguir. Busca-se também realizar um experimento a partir de uma representação vetorial de palavras (*Word Embeddings*, GloVe). Na construção do modelo *bag-of-words*, foi verificada a presença do termo ao invés da frequência, isto é, uma matriz construída de forma binária, pois essa estratégia é mais eficaz para classificação do sentimento [15]. As *features* foram utilizadas como unigrama, bigrama e unigrama+bigrama. Para os testes foram utilizados 3 algoritmos de classificação: SVM (*Support Vector Machine*), LR (*Logistic Regression*) e RF (*Random Forest*), sendo que o RF foi somente utilizado nos métodos *ensembles*. Escolhemos esses algoritmos por serem frequentemente utilizados em trabalhos de análise de sentimentos, como em [10], [35], [42].

A seguir será explicado todo o processo de classificação desde a etapa de pré-processamento até a validação dos dados. Também será detalhado como cada método foi aplicado ao modelo *bag-of-words*, isto é, cada método gera uma matriz que é concatenada à *bag-of-words*. Por fim, será mostrado como o GloVe foi combinado a *bag-of-words* e ao *Opinion Lexicon* [43].

Pré-processamento: Essa etapa consiste em eliminar ruídos e termos que não possuem significado semântico como a remoção de links, números, caracteres especiais e *stop words*.

Foi aplicada também a padronização dos *tweets* em minúsculo e o *stemming*. O *stemming* não necessariamente reduz ao radical, mas sim a uma forma canônica. Isto é, segundo [44], *stemming* é a redução das palavras em seu morfema. Um morfema (*stem*, em inglês), ou radical, é a menor parte com significado de uma palavra, portanto, no processo de *stemming*, palavras como casa, casas, casinhas e casarão resultam no mesmo morfema: cas. Entende-se que o termo canônico está mais associado à lematização, que ao ser aplicada ao exemplo anterior resultaria em uma palavra com significado válido, ou seja: casa. Pode-se dizer também que no processo de *stemming* os prefixos e sufixos são retirados [45].

Dicionários léxicos: Para esta técnica foram utilizado três dicionários léxicos: *Opinion Lexicon*¹⁶ [43], *SenticNet*¹⁷ [6] e *SemEval2015-Lexicon*¹⁸ [12].

- *Opinion Lexicon*: possui 4.783 léxicos positivos e 2.006 negativos. Para esse método utilizamos a estratégia proposta por Mohammad, Kiritchenko e Zhu em [20], foi feita uma contagem de léxicos positivos e negativos presentes em cada *tweet*. Se o número de palavras positivas for maior que o número de palavras negativas, então, o *tweet* é positivo, caso contrário o *tweet* é negativo. Se houver empate entre

palavras positivas e negativas, o *tweet* é neutro.

- *SenticNet*: conta com 50.000 palavras classificadas como positivas e negativas. Assim como no *Opinion Lexicon* [43], para o *SenticNet* aproveitamos a mesma estratégia descrita anteriormente.
- *SemEval2015-Lexicon*: possui 1515 léxicos, cada palavra possui uma pontuação (um número real). Nessa etapa soma-se os pontos de palavras positivas e negativas, e então verifica-se o resultado final, como mostra a Equação 1:

$$n = \sum_{i=1}^N K_i \quad (1)$$

onde K_i representa a pontuação de cada *tweet*. Se $n > 0$, então o *tweet* é positivo, se $n < 0$, então o *tweet* é negativo, se $n = 0$, então o *tweet* é neutro.

TABELA II

UMA REPRESENTAÇÃO DE TWEETS BASEADOS EM DICIONÁRIO LÉXICO.

	positivo	negativo	neutro
<i>tweet</i> ₁	1	0	0
<i>tweet</i> ₂	0	0	1
<i>tweet</i> ₃	0	1	0
...
<i>tweet</i> _n	1	0	0

Emoticons: Utilizamos o *Emoticon Sentiment Lexicon*¹⁹ [46], que detém 476 *emoticons*, distribuídos em 179 com sentimento positivo, 278 com sentimento negativo e 20 com sentimento neutro. Nesse caso, a estratégia é verificar se o *tweet* possui um *emoticon*. No caso de haver *emoticon* positivo, então o *tweet* é positivo. Já se o *emoticon* encontrado for negativo, o *tweet* é dado como negativo. Por fim, se detectado um *emoticon* neutro, o *tweet* será neutro. A Tabela III mostra alguns dos *emoticons* que expressam sentimentos.

TABELA III

EXEMPLOS DE EMOTICONS DE SENTIMENTO.

positivos	negativos	neutros
:), :'), :-)	:-*(, :-c, :/	(o;, :l, :-O, ;)
:-*, :-}, :}, :D	#-(, :#, #(;o), <:}, }:-)

Part-of-Speech: Trata-se de uma técnica que permite categorizar cada palavra na respectiva classe sintática, como: verbo, pronome, advérbio, entre outros. A Tabela IV

¹⁶<https://www.cs.uic.edu/~liub/>.¹⁷<http://sentic.net/>.¹⁸<http://www.saifmohammad.com/WebPages/SCL.html>.¹⁹<http://saifmohammad.com/WebPages/lexicons.html>.

mostra alguns exemplos de *tags*²⁰ [47]. Muitos pesquisadores aplicaram *part-of-speech* em seus trabalhos [33]–[35]. Nesse método foi aplicado o pacote de *Stanford*²¹ [48]. Nessa fase, utilizamos uma técnica chamada *tokenização*, que divide um *tweet* em palavras e pontuações, então atribuímos a cada *token* uma *tag*. Após a rotulação de cada *tweet*, é feita uma contagem de cada *tag* em cada *tweet*, desse modo, criamos uma matriz que foi concatenada ao modelo de *bag-of-words*.

TABELA IV
EXEMPLOS DE PART-OF-SPEECH TAGGER DE STANDARD.

Tag	Descrição	Exemplo
CC	conjunção	<i>e, ou, mas</i>
JJ	adjetivo	<i>bom, brilhante</i>
NN	substantivo	<i>felicidade, livro</i>
RB	advérbio	<i>hoje, já, sim</i>
VB	verbo	<i>falou, falaria</i>
SYM	símbolo	<i>+, %, #</i>

TABELA V
UMA REPRESENTAÇÃO DE TWEETS BASEADA EM PART-OF-SPEECH.

	CC	JJ	VB	...	NN	classe
<i>tweet₁</i>	0	2	0	...	0	positivo
<i>tweet₂</i>	1	0	3	...	0	negativo
<i>tweet₃</i>	0	2	0	...	0	neutro
...
<i>tweet_n</i>	1	0	2	...	0	positivo

Veja na Tabela VI como fica um exemplo prático na seguinte frase: *Infelizmente este computador é muito caro!*.

TABELA VI
UME EXEMPLO PRÁTICO DE UMA CLASSIFICAÇÃO SINTÁTICA.

RB	JJ	NN	VB	SYM	NPN	classe
2	1	1	1	1	1	negativo

Note que neste exemplo temos: 2 advérbios, **infelizmente** e **muito**; 1 adjetivo, **caro**; 1 substantivo, **computador**; 1 verbo, **é**; 1 símbolo, **!**; e 1 pronome, **este**.

Os métodos baseados em dicionários léxicos e *emoticons* geram uma matriz binária, como mostra a Tabela II. O método baseado em *part-of-speech* também gera um matriz, porém não binária, como apresenta a Tabela V. Posteriormente, essas matrizes são concatenadas com a *bag-of-words*, como mostra a Figura 3. Por fim, o algoritmo recebe como entrada a matriz resultante juntamente com as classes dos *tweets*.

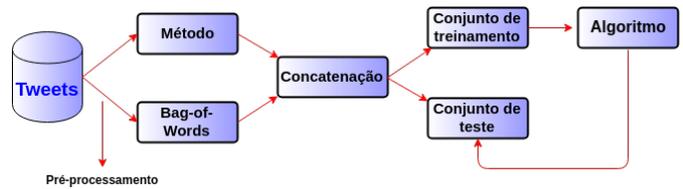


Fig. 3. Representando como cada método é concatenado, individualmente, com a *bag-of-words*.

Combinações de métodos: As mesmas estratégias descritas anteriormente foram mantidas, porém, combinamos métodos na construção do modelo de classificação. As combinações foram realizadas da seguinte forma: *emoticons* + léxico (neste caso o *Opinion Lexicon*), os 3 dicionários léxicos, e por último, *part-of-speech* + *emoticons* + léxico (novamente *Opinion Lexicon*). Os métodos híbridos estão sendo explorados na literatura, como em [49]–[52]. O modo como é feita a combinação pode ser observado na Figura 4.

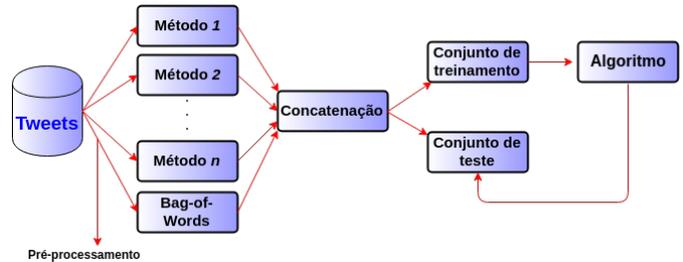


Fig. 4. Representando como é feita a combinação de métodos e a concatenação com a *bag-of-words*.

Ensembles: Métodos *ensembles* usam múltiplos classificadores para resolver o mesmo problema. A ideia por trás desse método, é que uma coleção de diferentes classificadores podem oferecer informações complementares com relação aos padrões que serão classificados, melhorando a eficácia de todo o processo de aprendizado [10]. Nessa técnica os termos foram representados somente por unigrama+bigrama. Existem diversas formas de combinar classificadores a fim de obter um *ensemble*, uma dessas formas segue a regra de voto majoritário, como pode ser visto na Figura 5. O leitor pode conferir outros estudos com *ensembles* em [53]–[55]. O modelo de *ensemble* segue na Figura 6.

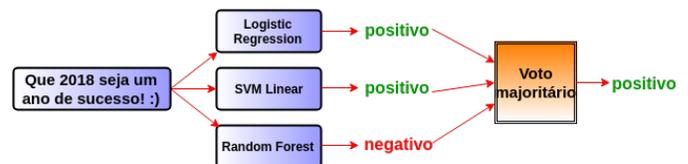
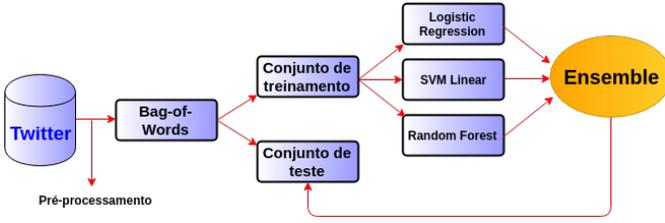


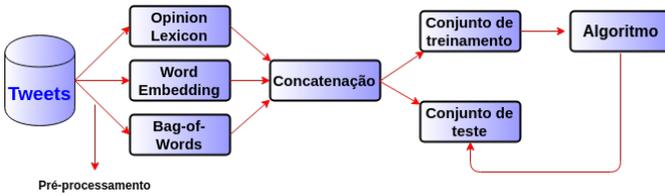
Fig. 5. Neste caso, a maioria dos classificadores concordam que a classe é positiva.

²⁰As *tags* representam a classe sintática das palavras.

²¹<https://nlp.stanford.edu/>.


 Fig. 6. Uma representação de como aplica-se o método *ensemble*.

Word Embedding: Como mencionado anteriormente, no intuito de minimizar os problemas de esparsidade dos dados e definição de contexto, selecionamos um modelo pré-treinado do GloVe [38] com 25, 50, 100 e 200 dimensões. A matriz gerada a partir da média dos vetores de cada palavra foi concatenada ao modelo *bag-of-words* e também ao *Opinion Lexicon*. O processo desse modelo pode ser visto na Figura 7.


 Fig. 7. Uma representação de como é construído uma configuração baseada em dicionário léxico, *bag-of-words* e *word embeddings* (GloVe).

Avaliação e validação: Para a avaliação do desempenho do classificador e do comportamento do modelo, primeiro é preciso entender o que é Acurácia (*Accuracy*), Precisão (*Precision*), Revocação (*Recall*) e a Medida F (*F1-measure*). Mas antes, vamos apresentar algumas definições importantes [36] estendidas para o problema de três classes consideradas (positiva, negativa e neutra), então temos para cada classe C_i [56]:

- *True positive* (TP_i): verdadeiro positivo para C_i .
- *True negative* (TN_i): verdadeiro negativo para C_i .
- *False positive* (FP_i): falso positivo para C_i .
- *False negative* (FN_i): falso negativo para C_i .

Para cada classe C_i , temos: a Precisão é o somatório de TP_i , dividido por $(TP_i + FP_i)$, tudo dividido pelo número k de classes, como pode ser observado na Equação 2. Já a Revocação é o somatório de TP_i , dividido por $(TP_i + FN_i)$, dividido pelo número k de classes, como ilustra a Equação 3. Enquanto a Medida F consiste na média harmônica entre a precisão e a revocação. Com essa informação podemos avaliar a performance do classificador com um indicador apenas, como mostra a Equação 4. Por fim, tem-se a acurácia que demonstra como o classificador se saiu de uma maneira geral, pois mede a quantidade de acertos sobre o todo. O cálculo da acurácia é apresentado na Equação 5.

$$precision = \frac{\sum_{i=1}^K \frac{TP_i}{TP_i + FP_i}}{K} \quad (2)$$

$$recall = \frac{\sum_{i=1}^K \frac{TP_i}{TP_i + FN_i}}{K} \quad (3)$$

$$f\text{-measure} = \frac{(\beta^2 + 1) * Precision * Recall}{\beta^2 * Precision + Recall} \quad (4)$$

$$accuracy = \frac{\sum_{i=1}^K \frac{TP_i + TN_i}{TP_i + FN_i + FP_i + TN_i}}{K} \quad (5)$$

Para validar os modelos de cada experimento, aplicamos a técnica de validação cruzada [57]. Esse procedimento consiste em dividir a base de dados em k partes, essas partes são chamadas de *folds*. Uma dessas partes é escolhida para testar o modelo, enquanto o restante é utilizado na fase de treinamento, isso é feito repetidamente até que o modelo seja treinado e testado com todas as partes. Nos experimentos realizados na base de dados *Sanders* utilizamos 10 *folds*. Esse processo avalia a capacidade de generalização de um modelo a partir de um conjunto de dados. Devido a essa generalização, os problemas de variância nos dados são minimizados. Na *HCR* não houve validação cruzada, pois esta conta com uma base de *tweets* exclusivamente para teste. Confira na Figura 8 o processo de validação cruzada.

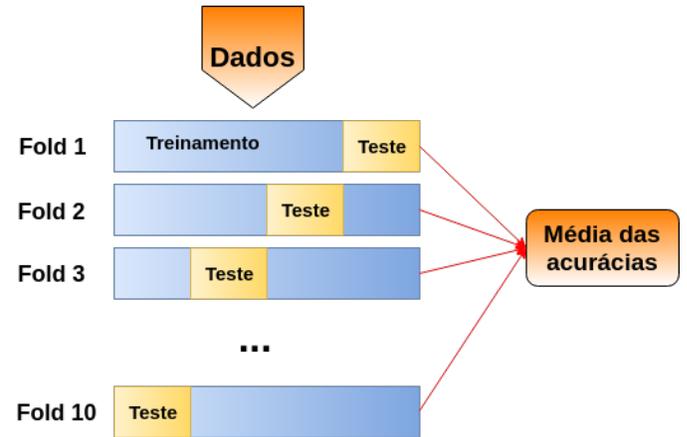


Fig. 8. Processo de validação cruzada.

VI. RESULTADOS

DIVERSOS experimentos foram realizados em cada *dataset* e avaliados seguindo as métricas explicadas na seção anterior. Idealmente, uma configuração possui classificação perfeita quando atinge o valor máximo de 100% de acurácia [58].

A. Resultados Sanders

A Tabela VII apresenta os resultados para cada S_i configurações, e a Tabela VIII compara os melhores resultados. Enquanto que a Tabela IX mostra os resultados quando se combinam classificadores (*ensembles*). Por fim, a Tabela X aponta os resultados com *word embeddings*.

TABELA VII
RESULTADOS DA BASE DE DADOS SANDERS

S1: Aprendizado de Máquina					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,89	69,91	65,64	67,50
Unigrama	LR	77,54	74,55	60,72	65,08
Bigrama	SVM	74,83	69,56	55,56	59,77
Uni+Big	LR	78,10	76,92	60,50	65,38
S2: POS tagger Stanford					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,14	68,54	65,35	66,78
Unigrama	LR	77,13	72,56	61,60	65,31
Bigrama	SVM	75,50	68,33	59,79	62,85
Uni+Big	LR	77,11	72,78	60,69	64,54
S3: Opinion Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,92	69,51	66,49	67,85
Unigrama	LR	78,02	73,71	63,51	67,18
Bigrama	SVM	75,93	70,69	59,96	63,46
Uni+Big	LR	79,09	76,20	64,22	68,41
S4: SemEval-2015 Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,25	68,40	64,11	65,95
Unigrama	LR	77,27	73,65	60,71	64,71
Bigrama	SVM	74,34	71,75	53,21	57,21
Uni+Big	LR	77,78	75,09	60,71	65,17
S5: SenticNet					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,11	68,47	64,26	66,06
Unigrama	LR	76,27	71,94	59,07	63,07
Bigrama	SVM	73,27	71,09	50,46	54,36
Uni+Big	LR	77,24	75,61	59,34	64,05
S6: Emoticons Sentiment Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	75,71	67,83	63,65	65,44
Unigrama	LR	76,01	72,00	58,56	62,60
Bigrama	SVM	73,03	71,26	49,86	53,70
Uni+Big	LR	78,07	77,61	60,42	65,41
S7: Opinion Lexicon + Emoticons					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,01	68,47	64,26	66,06
Unigrama	LR	78,53	71,94	59,07	63,07
Bigrama	SVM	75,74	71,09	50,46	54,36
Uni+Big	LR	78,96	75,61	59,34	64,05
S8: Opinion Lexicon + SemEval-2015 + SenticNet					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,68	68,97	66,30	67,52
Unigrama	LR	78,13	74,08	63,93	63,06
Bigrama	SVM	75,42	69,87	58,78	62,35
Uni+Big	LR	78,31	74,44	63,45	67,30
S9: Opinion Lexicon + Emoticons + POS tagger					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	76,84	69,34	66,63	67,63
Unigrama	LR	78,15	72,90	64,19	64,43
Bigrama	SVM	76,19	69,91	61,10	64,28
Uni+Big	LR	78,96	74,81	65,21	68,77

Os resultados apontam S3 como o método que obteve o melhor resultado, com 79,09% de acurácia. E nota-se a predominância do classificador LR, bem como atributos agregados (unigrama + bigrama), sugerindo que esta combinação gera melhores resultados quando expostos aos *tweets* da base de dados em questão (Sanders). Somente o método S2 classificou melhor os *tweets* utilizando unigrama, e nenhum método teve o melhor resultado utilizando bigrama.

Neste experimento foi aplicado dois classificadores de base, o *Support Vector Machine* e o *Logistic Regression*.

TABELA VIII
TABELA COMPARATIVA ENTRE OS MELHORES RESULTADOS DOS MÉTODOS PROPOSTOS.

Método	Atributo	Algoritmo	Acurácia (%)
S1	Uni+Big	LR	78,10
S2	Unigrama	LR	77,13
S3	Uni+Big	LR	79,09
S4	Uni+Big	LR	77,78
S5	Uni+Big	LR	77,24
S6	Uni+Big	LR	78,07
S7	Uni+Big	LR	78,96
S8	Uni+Big	LR	78,31
S9	Uni+Big	LR	78,96

TABELA IX
RESULTADOS DOS MODELOS ENSEMBLES PARA A SANDERS.

	Acc. (%)		P (%)	R (%)	F1 (%)
	EN	EXP1			
S1	78,05	< 78,10	75,66	61,23	65,78
S2	77,83	> 77,13	74,61	61,62	65,76
S3	78,21	< 79,09	74,95	63,08	67,15
S4	77,38	< 77,78	75,26	60,13	64,65
S5	76,97	< 77,24	75,53	58,22	63,00
S6	76,95	< 78,07	75,76	58,31	63,11
S7	78,07	< 78,96	74,88	62,32	66,50
S8	78,10	< 78,31	74,93	62,62	66,75
S9	78,37	< 78,96	74,40	63,83	67,50

Os resultados deste experimento mostram que na maioria dos métodos, um classificador de base obtém uma acurácia melhor, mesmo que a diferença de resultado seja pequena. Somente um classificador *ensemble* foi levemente superior, que é o caso do método S2.

Os algoritmos que fizeram parte deste experimento foram: LR (*Logistic Regression*), SVM (*Support Vector Machine*) e RF (*Random Forest*).

TABELA X
RESULTADOS OBTIDOS COM WORD EMBEDDING (GLOVE) PARA OS
TWEETS DA SANDERS.

Dimensão	Acc. (%)	P (%)	R (%)	F1 (%)
25	78,56	74,53	63,36	67,25
50	78,77	75,17	63,68	67,66
100	79,15	75,83	64,15	68,26
200	79,36	75,82	64,89	68,85

Com a premissa de que *word embeddings* reduz o problema de espasidade, o modelo proposto que combina *bag-of-words* e *word embeddings* com dicionário léxico (*Opinion Lexicon*) obteve 79,36% de acurácia. Comparando com o modelo anterior sem *word embeddings* (S3, Tabela VII), o qual obteve 79,09% de acurácia. Esse resultado não demonstra uma diferença significativa, embora levante uma questão sobre qual seria o resultado somente aplicando *word embeddings*? Nesse experimento o classificador aplicado foi o *Logistic Regression*.

B. Resultados HCR

A Tabela XII apresenta os resultados obtidos referentes a cada H_i configurações, e a Tabela XI compara os melhores resultados. Já a Tabela XIII mostra os resultados quando se combinam classificadores (*ensembles*). Finalmente, a Tabela XIV aponta os resultados referentes a combinação de *word embeddings* com *bag-of-words* e *Opinion Lexicon*.

TABELA XI
TABELA COMPARATIVA ENTRE OS MELHORES RESULTADOS DOS
MÉTODOS PROPOSTOS.

Método	Atributo	Algoritmo	Acurácia (%)
H1	Uni+Big	LR	65,44
H2	Uni+Big	LR	63,61
H3	Unigrama	LR	69,11
H4	Uni+Big	LR	66,36
H5	Uni+Big	LR	64,83
H6	Unigrama	LR	62,69
H7	Unigrama	LR	64,83
H8	Unigrama	SVM	64,53
H9	Uni+Big	LR	63,30

Os resultados da Tabela XI mostram, novamente, que o método H3 (*Opinion Lexicon*) obteve o melhor resultado,

TABELA XII
RESULTADOS DA BASE DE DADOS HCR

H1: Aprendizado de Máquina					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	60,24	57,94	56,08	56,53
Unigrama	LR	64,22	62,33	57,17	57,79
Bigrama	SVM	60,55	55,99	48,57	48,87
Uni+Big	LR	65,44	62,83	59,36	60,47
H2: POS tagger Stanford					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	59,63	54,88	54,03	54,36
Unigrama	LR	63,30	60,36	57,47	58,43
Bigrama	SVM	59,63	53,50	52,79	52,94
Uni+Big	LR	63,61	62,03	57,97	59,29
H3: Opinion Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	62,39	59,77	58,35	58,92
Unigrama	LR	69,11	66,78	62,67	63,90
Bigrama	SVM	62,08	60,87	54,71	55,46
Uni+Big	LR	67,58	67,63	60,03	60,90
H4: SemEval-2015 Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	62,69	60,37	58,54	58,95
Unigrama	LR	65,44	63,54	58,17	59,53
Bigrama	SVM	58,10	57,12	51,79	51,98
Uni+Big	LR	66,36	62,17	55,66	57,29
H5: SenticNet					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	61,16	57,95	57,92	57,93
Unigrama	LR	60,55	57,61	53,56	54,78
Bigrama	SVM	59,33	55,08	50,82	50,97
Uni+Big	LR	64,83	62,99	58,00	59,34
H6: Emoticons Sentiment Lexicon					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	61,16	58,00	56,93	56,99
Unigrama	LR	62,69	63,32	57,03	58,86
Bigrama	SVM	58,72	57,00	49,08	49,22
Uni+Big	LR	62,08	64,84	54,89	56,20
H7: Opinion Lexicon + Emoticons					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	60,55	59,97	57,29	58,04
Unigrama	LR	64,83	62,31	58,52	59,57
Bigrama	SVM	59,02	54,82	51,85	51,65
Uni+Big	LR	62,39	61,00	55,83	56,44
H8: Opinion Lexicon + SemEval-2015 + SenticNet					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	64,53	61,46	61,59	61,52
Unigrama	LR	62,69	58,15	56,38	57,08
Bigrama	SVM	60,86	57,58	53,09	54,25
Uni+Big	LR	62,08	58,06	55,03	55,89
H9: Opinion Lexicon + Emoticons + POS tagger					
Atributo	Algoritmo	Acc. (%)	P (%)	R (%)	F1 (%)
Unigrama	SVM	62,69	61,25	59,60	60,23
Unigrama	LR	59,94	57,31	54,74	55,05
Bigrama	SVM	62,39	63,15	58,17	59,05
Uni+Big	LR	63,30	60,43	59,06	59,58

com 69,11% de acurácia. Neste experimento é possível observar que atributos como unigrama estiveram mais presentes, enquanto que bigrama não demonstrou-se ter característica preditiva em comparação a outras formas representativas.

Repetidamente, o algoritmo *Logistic Regression* mostra-se presente nos melhores resultados. Enquanto que o classificador *Support Vector Machine* apresentou-se melhor somente no método H8, este que obteve 64,53% de acurácia.

Desse modo, os resultados sugerem que tais configurações provêm bons resultados diante dos *tweets* (HCR).

TABELA XIII
RESULTADOS DOS MODELOS ENSEMBLES PARA HCR.

	Acc. (%)		P (%)	R (%)	F1 (%)
	EN	EXP1			
H1	65,44 =	65,44	64,79	57,90	59,45
H2	60,24 <	63,61	60,22	52,50	53,88
H3	63,91 <	69,11	62,94	56,69	57,47
H4	61,77 <	66,36	58,17	53,40	54,56
H5	64,22 <	64,83	62,68	55,45	56,82
H6	64,53 >	62,69	62,04	55,68	56,89
H7	65,14 >	64,83	62,93	58,06	59,13
H8	65,44 >	64,53	61,89	59,29	59,97
H9	64,53 >	63,30	63,48	57,28	58,62

Comparando com os melhores resultados do primeiro experimento da HCR com os resultados obtidos com classificadores *ensembles*, nota-se uma pequena melhora nos resultados. Embora ainda não ser possível determinar que um classificador *ensemble* será sempre melhor, visto que um único classificador de base consegue obter acurácia melhor, que é o caso dos métodos H2, H3, H4 e H5.

Os algoritmos que fizeram parte deste experimento foram: LR (*Logistic Regression*), SVM (*Support Vector Machine*) e RF (*Random Forest*).

TABELA XIV
RESULTADOS OBTIDOS COM WORD EMBEDDING (GLOVE) PARA OS TWEETS DA HCR.

Dimensão	Acc. (%)	P (%)	R (%)	F1 (%)
25	65,68	63,49	60,29	61,14
50	68,22	70,31	63,34	65,34
100	62,29	58,92	55,56	55,91
200	66,53	64,58	60,73	61,61

Visando minimizar o problema de esparsidade, foi proposto um modelo combinando *bag-of-words* e *word embeddings* com dicionário léxico (*Opinion Lexicon*). Esse modelo obteve 68,22% de acurácia, comparando com o modelo anterior sem *word embeddings* (H3, Tabela XII), o qual obteve 69,11% de acurácia. Nesse experimento o classificador aplicado foi o *Logistic Regression*.

C. Comparação entre léxicos

Em resposta a uma questão levantada, em uma listagem constatou-se que existem 243 palavras em comum

entre os léxicos presentes neste artigo. Dessas palavras, 14 contradizem-se em seu sentimento, como pode ser visto na Tabela XV.

A Tabela XVI e a Tabela XVII mostram o comparativo entre os dicionários léxicos para os *tweets* da Sanders e HCR, respectivamente.

TABELA XV
PALAVRAS QUE POSSUEM SENTIMENTOS CONTRADITÓRIOS.

Palavras	SemEval2015	SenticNet	Opinion Lexicon
abysmal	negativo	positivo	negativo
enough	negativo	positivo	negativo
flirt	positivo	positivo	negativo
funny	positivo	positivo	negativo
hang	negativo	positivo	negativo
hot	positivo	negativo	positivo
joke	positivo	positivo	negativo
layghable	positivo	negativo	negativo
mediocrity	negativo	positivo	negativo
miss	positivo	negativo	negativo
pleasantly	positivo	negativo	positivo
tired	negativo	positivo	negativo
waste	negativo	positivo	negativo
work	negativo	negativo	positivo

TABELA XVI
TABELA COMPARATIVA REFERENTE AOS DICIONÁRIOS LÉXICOS APLICADOS NA BASE DE DADOS SANDERS.

Método	Atributo	Algoritmo	Acurácia (%)
Opinion Lexicon	Uni+Big	LR	79,09
SemEval-2015	Uni+Big	LR	77,78
SenticNet	Uni+Big	LR	77,24

TABELA XVII
TABELA COMPARATIVA REFERENTE AOS DICIONÁRIOS LÉXICOS APLICADOS NA BASE DE DADOS HCR.

Método	Atributo	Algoritmo	Acurácia (%)
Opinion Lexicon	Uni+Big	LR	69,11
SemEval-2015	Uni+Big	LR	66,36
SenticNet	Uni+Big	LR	64,83

Os resultados mostram a superioridade do *Opinion Lexicon* em ambos conjuntos de dados, com 79,09% de acurácia para Sanders e 69,11% de acurácia para HCR. No entanto, a diferença de resultados em relação aos outros dicionários léxicos é pequena, não passando de 2% para Sanders e 4,5% para HCR, aproximadamente. Com essa diferença se faz necessário realizar um estudo detalhado para compreender o desempenho dos dicionários. Tampouco pode-se afirmar que a distinção dos sentimentos das palavras levantadas influenciam os resultados.

De fato, é importante ressaltar a predominância de atributos agregados (unigrama + bigrama), bem como

o classificador presente nos experimentos, o *Logistic Regression*. Mostrando que essa combinação obtém melhores resultados diante das particularidades dos *tweets*.

VII. CONSIDERAÇÕES FINAIS

DIANTE da importância dos métodos de classificação de sentimentos, o presente trabalho propôs um estudo, no qual diferentes métodos da literatura foram explorados frente duas bases de dados com contextos distintos, Sanders e HCR.

Os métodos selecionados são baseados em aprendizado de máquina, orientação sintática (*part-of-speech*), dicionário léxico, *emoticons*, *ensembles* e *word embeddings*. Além disso, combinações entre métodos foram feitas com o intuito de buscar melhorar a predição do *tweets*.

Como classificadores de base, 3 algoritmos foram escolhidos: LR (*Regression Logistic*), SVM (*Support Vector Machine*) e RF (*Random Forest*), sendo que o RF foi utilizado apenas em modelos *ensembles*, o qual visa agregar classificadores.

O primeiro experimento (Tabela VII) mostrou que métodos baseados em dicionários léxicos produzem resultados melhores, que é o caso dos métodos S3 (*Opinion Lexicon* da base Sanders) e H3 (*Opinion Lexicon* da base HCR), onde S3 obteve 79,09% de acurácia e H3 obteve 69,11%.

Com relação aos *ensembles* (Tabela IX) apenas o método S2 (*part-of-speech*) obteve uma melhora sem muita expressão, em comparação aos experimentos da Tabela VII. Já nos *ensembles* da HCR (Tabela XIII), 4 métodos também obtiveram uma melhora pouco expressiva, foram eles: H6, H7, H8 e H9.

Além disso, os experimentos com *word embeddings* (GloVe) não demonstraram grandes avanços, onde o melhor resultado para Sanders foi 79,36% de acurácia, e para HCR foi 68,22% de acurácia.

Com o intuito de melhorar os resultados, pretendemos como trabalhos futuros aplicar algoritmos de regressão, os quais pouco se tem explorado [59], [60]. A hipótese é que a análise de sentimentos pode ser melhor abordada quando considerada de forma “não rígida” (a classificação não fica restrita somente às classes positiva, negativa e neutra), isto é, quando a opinião extraída está em um *rating* de valores (por exemplo [0, 1]) discriminaria melhor o seu real sentimento – um texto com um *rating* de sentimento = 0.8 é mais positivo que um texto com *rating* de sentimento = 0.6. Desejamos também realizar um estudo mais profundo sobre *word embeddings*, explorando outros *word vectors*, como Word2vec [7] e FastText [61].

REFERÊNCIAS

- [1] B. Pang e L. Lee, “Opinion mining and sentiment analysis”, *Found. Trends Inf. Retr.*, vol. 2, n° 1-2, pp. 1–135, jan. de 2008, ISSN: 1554-0669. DOI: 10.1561/1500000011. endereço: <http://dx.doi.org/10.1561/1500000011>.
- [2] M. Speriosu, N. Sudan, S. Upadhyay e J. Baldrige, “Twitter polarity classification with label propagation over lexical links and the follower graph”, em *Proceedings of the First Workshop on Unsupervised Learning in NLP*, sér. EMNLP ’11, Edinburgh, Scotland: Association for Computational Linguistics, 2011, pp. 53–63, ISBN: 978-1-937284-13-8. endereço: <http://dl.acm.org/citation.cfm?id=2140458.2140465>.
- [3] Y. H. Hassan Saif, Miriam Fernandez e H. Alani, “Evaluation datasets for twitter sentiment analysis: A survey and a new dataset, the sts-gold”, *First ESSEM workshop*, 2013.
- [4] B. Liu, *Sentiment analysis and opinion mining*. Morgan & Claypool Publishers, 2012, ISBN: 1608458849, 9781608458844.
- [5] T. Hardeniya e D. A. Borikar, “An approach to sentiment analysis using lexi cons with comparative analysis of different techniques”, *IOSR Journals*, vol. 18, n° 3, pp. 53–57, 2016.
- [6] E. Cambria, S. Poria, R. Bajpai e B. W. Schuller, “Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives”, em *COLING*, 2016.
- [7] T. Mikolov, K. Chen, G. Corrado e J. Dean, “Efficient estimation of word representations in vector space”, *ArXiv preprint arXiv:1301.3781*, 2013.
- [8] Y. Mansar, L. Gatti, S. Ferradans, M. Guerini e J. Staiano, “Fortia-fbk at semeval-2017 task 5: Bullish or bearish? inferring sentiment towards brands from financial news headlines”, *ArXiv preprint arXiv:1704.00939*, 2017.
- [9] L. Rotim, M. Tutek e J. Šnajder, “Takelab at semeval-2017 task 5: Linear aggregation of word embeddings for fine-grained sentiment analysis of financial news”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 866–871.
- [10] N. F. F. SILVA, “Análise de sentimentos em textos curtos provenientes de redes sociais”, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2016.
- [11] H. Saif, Y. He e H. Alani, “Alleviating data sparsity for twitter sentiment analysis”, em *Workshop of Making Sense of Microposts co-located with WWW 2012*, 2012. endereço: http://ceur-ws.org/Vol-838/paper_01.pdf.
- [12] S. Kiritchenko, X. Zhu e S. M. Mohammad, “Sentiment analysis of short informal texts”, *Journal of Artificial Intelligence Research*, vol. 50, pp. 723–762, 2014.
- [13] H. Saif, M. Fernandez, Y. He e H. Alani, “On stopwords, filtering and data sparsity for sentiment analysis of twitter”, inglês, em *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, N. C. (Chair), K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk e S. Piperidis, eds.,

- Reykjavik, Iceland: European Language Resources Association (ELRA), maio de 2014.
- [14] K. Dashtipour, S. Poria, A. Hussain, E. Cambria, A. Y. A. Hawalah, A. Gelbukh e Q. Zhou, “Multilingual sentiment analysis: State of the art and independent comparison of techniques”, *Cognitive Computation*, vol. 8, n° 4, pp. 757–771, ago. de 2016.
- [15] B. Pang, L. Lee e S. Vaithyanathan, “Thumbs up?: Sentiment classification using machine learning techniques”, em *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, sér. EMNLP '02, Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 79–86. DOI: 10.3115/1118693.1118704. endereço: <https://doi.org/10.3115/1118693.1118704>.
- [16] L. Rokach, “Ensemble-based classifiers”, *Artif. Intell. Rev.*, vol. 33, n° 1-2, pp. 1–39, fev. de 2010, ISSN: 0269-2821. DOI: 10.1007/s10462-009-9124-7. endereço: <http://dx.doi.org/10.1007/s10462-009-9124-7>.
- [17] M. Araújo, P. Gonçalves e F. Benevenuto, “Measuring sentiments in online social networks”, em *Proceedings of the 19th Brazilian Symposium on Multimedia and the Web*, sér. WebMedia '13, Salvador, Brazil: ACM, 2013, pp. 97–104, ISBN: 978-1-4503-2559-2. DOI: 10.1145/2526188.2526196. endereço: <http://doi.acm.org/10.1145/2526188.2526196>.
- [18] S. Baccianella, A. Esuli e F. Sebastiani, “Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining”, em *LREC*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odiijk, S. Piperidis, M. Rosner e D. Tapias, eds., European Language Resources Association, 2010, ISBN: 2-9517408-6-7. endereço: <http://nmis.isti.cnr.it/sebastiani/Publications/LREC10.pdf>.
- [19] J. C. Reis, P. Gonçalves, M. Araújo, A. C. Pereira e F. Benevenuto, “Uma abordagem multilíngue para análise de sentimentos”, *IV Brazilian*, 2015.
- [20] S. M. Mohammad, S. Kiritchenko e X. Zhu, “Nrcanada: Building the state-of-the-art in sentiment analysis of tweets”, *CoRR*, vol. abs/1308.6242, 2013. endereço: <http://arxiv.org/abs/1308.6242>.
- [21] M. Fouad, T. Gharib e A. Mashat, *Efficient twitter sentiment analysis system with feature selection and classifier ensemble*, jan. de 2018.
- [22] P. Chaovalit e L. Zhou, “Movie review mining: A comparison between supervised and unsupervised classification approaches”, em *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*, IEEE, 2005, pp. 112c–112c.
- [23] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan e N. A. Smith, “Part-of-speech tagging for twitter: Annotation, features, and experiments”, em *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, sér. HLT '11, Portland, Oregon: Association for Computational Linguistics, 2011, pp. 42–47, ISBN: 978-1-932432-88-6.
- [24] J.-S. R. Jang, C.-T. Sun e E. Mizutani, “Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence”, 1997.
- [25] G. A. Miller, “Wordnet: A lexical database for english”, *Commun. ACM*, vol. 38, n° 11, pp. 39–41, nov. de 1995, ISSN: 0001-0782. DOI: 10.1145/219717.219748. endereço: <http://doi.acm.org/10.1145/219717.219748>.
- [26] S. Rosenthal, A. Ritter, P. Nakov e V. Stoyanov, “Semeval-2014 task 9: Sentiment analysis in twitter”, em *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 2014, pp. 73–80.
- [27] P. P. Balage Filho, L. V. Avanço, T. A. S. Pardo, M. d. G. V. Nunes et al., “Nilc_usp: An improved hybrid system for sentiment analysis in twitter messages.”, em *International Workshop on Semantic Evaluation, 8th*, ACL Special Interest Group on the Lexicon-SIGLEX, 2014.
- [28] P. Kumar e U. C. Jaiswal, “A comparative study on sentiment analysis and opinion mining”, *International Journal of Engineering & Technology IJET*, vol. 8, n° 2, p. 938, 2016.
- [29] J. Staiano e M. Guerini, “Depechemood: A lexicon for emotion analysis from crowd-annotated news”, *ArXiv preprint arXiv:1405.1605*, 2014.
- [30] P. Saleiro, E. M. Rodrigues, C. Soares e E. Oliveira, “Feup at semeval-2017 task 5: Predicting sentiment polarity and intensity with financial word embeddings”, *ArXiv preprint arXiv:1704.05091*, 2017.
- [31] N. F. da Silva, E. R. Hruschka e E. R. Hruschka, “Tweet sentiment analysis with classifier ensembles”, *Decis. Support Syst.*, vol. 66, n° C, pp. 170–179, out. de 2014, ISSN: 0167-9236. DOI: 10.1016/j.dss.2014.07.003. endereço: <http://dx.doi.org/10.1016/j.dss.2014.07.003>.
- [32] K. K. Pawar e R. R. Deshmukh, “Twitter sentiment classification on sanders data using hybrid approach”, 2015.
- [33] F. Aisopos, G. Papadakis e T. Varvarigou, “Sentiment analysis of social media content using n-gram graphs”, em *Proceedings of the 3rd ACM SIGMM International Workshop on Social Media*, sér. WSM '11, Scottsdale, Arizona, USA: ACM, 2011, pp. 9–14, ISBN: 978-1-4503-0989-9. DOI: 10.1145/2072609.2072614. endereço: <http://doi.acm.org/10.1145/2072609.2072614>.
- [34] L. Barbosa e J. Feng, “Robust sentiment detection on twitter from biased and noisy data”, em *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, sér. COLING '10, Beijing, China: Association for Computational Linguistics, 2010, pp. 36–44. endereço: <http://dl.acm.org/citation.cfm?id=1944566.1944571>.

- [35] A. Go, R. Bhayani e L. Huang, “Twitter sentiment classification using distant supervision”, *CS224N Project Report, Stanford*, vol. 1, n^o 2009, p. 12, 2009.
- [36] P.-N. Tan, M. Steinbach e V. Kumar, *Introduction to data mining, (first edition)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005, ISBN: 0321321367.
- [37] Y. Bengio, R. Ducharme, P. Vincent e C. Jauvin, “A neural probabilistic language model”, *Journal of machine learning research*, vol. 3, n^o Feb, pp. 1137–1155, 2003.
- [38] J. Pennington, R. Socher e C. D. Manning, “Glove: Global vectors for word representation”, em *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. endereço: <http://www.aclweb.org/anthology/D14-1162>.
- [39] D. Ghosal, S. Bhatnagar, M. S. Akhtar, A. Ekbal e P. Bhattacharyya, “Iitp at semeval-2017 task 5: An ensemble of deep learning and feature based models for financial sentiment analysis”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 899–903.
- [40] A. Kumar, A. Sethi, M. S. Akhtar, A. Ekbal, C. Biemann e P. Bhattacharyya, “Iitpb at semeval-2017 task 5: Sentiment prediction in financial text”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 894–898.
- [41] N. J. Sanders, “Sanders-twitter sentiment corpus”, *Sanders Analytics LLC*, 2011.
- [42] T. Mullen e N. Collier, “Sentiment analysis using support vector machines with diverse information sources.”, em *EMNLP*, vol. 4, 2004, pp. 412–418.
- [43] M. Hu e B. Liu, “Mining and summarizing customer reviews”, em *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, sér. KDD '04, Seattle, WA, USA: ACM, 2004, pp. 168–177, ISBN: 1-58113-888-1. DOI: 10.1145/1014052.1014073. endereço: <http://doi.acm.org/10.1145/1014052.1014073>.
- [44] J. Perkins, *Python 3 text processing with nltk 3 cookbook*. Packt Publishing Ltd, 2014.
- [45] A. S. H. Basari, B. Hussin, I. G. P. Ananta e J. Zeniarja, “Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization”, *Procedia Engineering*, vol. 53, pp. 453–462, 2013.
- [46] A. Hogenboom, D. Bal, F. Frasinca, M. Bal, F. de Jong e U. Kaymak, “Exploiting emoticons in sentiment analysis”, em *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, ACM, 2013, pp. 703–710.
- [47] D. Jurafsky e J. H. Martin, *Speech and language processing*. Pearson London, 2014, vol. 3.
- [48] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard e D. McClosky, “The Stanford CoreNLP natural language processing toolkit”, em *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60. endereço: <http://www.aclweb.org/anthology/P/P14/P14-5010>.
- [49] N. Malandrakis, A. Kazemzadeh, A. Potamianos e S. Narayanan, “Sail: A hybrid approach to sentiment analysis”, em *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, vol. 2, 2013, pp. 438–442.
- [50] H. Thakkar e D. Patel, “Approaches for sentiment analysis on twitter: A state-of-art study”, *ArXiv preprint arXiv:1512.01043*, 2015.
- [51] P. Chauhan Ashish e D. K. Patel, “Sentiment analysis using hybrid approach: A survey”, *Int. Journal of Engineering Research and Applications www.ijera.com ISSN*, pp. 2248–9622,
- [52] A. Mudinas, D. Zhang e M. Levene, “Combining lexicon and learning based approaches for concept-level sentiment analysis”, em *Proceedings of the first international workshop on issues of sentiment discovery and opinion mining*, ACM, 2012, p. 5.
- [53] N. F. Da Silva, E. R. Hruschka e E. R. Hruschka, “Tweet sentiment analysis with classifier ensembles”, *Decision Support Systems*, vol. 66, pp. 170–179, 2014.
- [54] S. Rosenthal, N. Farra e P. Nakov, “Semeval-2017 task 4: Sentiment analysis in twitter”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 502–518.
- [55] P. Nakov, A. Ritter, S. Rosenthal, F. Sebastiani e V. Stoyanov, “Semeval-2016 task 4: Sentiment analysis in twitter”, em *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 1–18.
- [56] M. Sokolova e G. Lapalme, “A systematic analysis of performance measures for classification tasks”, *Information Processing & Management*, vol. 45, n^o 4, pp. 427–437, 2009.
- [57] I. H. Witten, E. Frank, M. A. Hall e C. J. Pal, *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.
- [58] P. Gonçalves, M. Araújo, F. Benevenuto e M. Cha, “Comparing and combining sentiment analysis methods”, em *Proceedings of the first ACM conference on Online social networks*, ACM, 2013, pp. 27–38.
- [59] Z. Nasim, “Iba-sys at semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 827–831. DOI: 10.18653/v1/S17-2140. endereço: <http://aclanthology.coli.uni-saarland.de/pdf/S/S17/S17-2140.pdf>.
- [60] M. Jiang, M. Lan e Y. Wu, “Ecnu at semeval-2017 task 5: An ensemble of regression algorithms with effective features for fine-grained sentiment

analysis in financial domain”, em *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 888–893. DOI: 10.18653/v1/S17-2152. endereço: <http://aclanthology.coli.uni-saarland.de/pdf/S/S17/S17-2152.pdf>.

- [61] A. Joulin, E. Grave, P. Bojanowski e T. Mikolov, “Bag of tricks for efficient text classification”, *CoRR*, vol. abs/1607.01759, 2016. arXiv: 1607.01759. endereço: <http://arxiv.org/abs/1607.01759>.