Exerting 2D-Space of Sentiment Lexicons with Machine Learning Techniques: A Hybrid Approach for Sentiment Analysis

Muhammad Yaseen Khan[1]
Mohammad Ali Jinnah University
Karachi, Pakistan

Khurum Nazir Junejo[2]
Ibex CX
Karachi, Pakistan

***Abstract*—Sentiment mining from the textual content on the web can give valuable insights for discernment, strategic decision making, targeted advertisement, and much more. Supervised machine learning (ML) approaches do not capture the sentiment inherent in the individual terms. Whereas the unsupervised sentiment lexicon (SL) based approaches lag behind ML approaches because of a bias they have towards one sentiment than the other. In this paper, we propose a hybrid approach that uses unsupervised sentiment lexicons to transform the term space into a two-dimensional sentiment space on which a discriminative classifier is trained in a supervised fashion. This hybrid approach yields higher accuracy, faster training, and lower memory footprint than the ML approaches. It is more suitable for scenarios where training data is scarce. We support our claim by reporting results on six social media datasets using five sentiment lexicons and four ML algorithms.**

***Keywords*—Hybrid approach; machine learning; sentiment analysis; sentiment lexicons; sentiment space; social media analysis**

I. Introduction

Humans are sentient beings that express emotions through sentiments. The behaviour of an individual is often guided by his (or her) emotions but can be studied by his (or her) sentiments. Sentiments are expressed through writings, speech, and actions. Recently, there is a drastic increase in usage of the online medium such as articles, blogs, e-shopping, online social networking (OSN) sites, e-newspapers, and magazines for expression of sentiments. Many people now present their analysis and stories in the forms of comments, tweets, reviews, and feedback on almost every aspect of life [1]. The automatic quantification of sentiments hidden within these texts can lead to many insights that can help in contextual advertisement [2], determine the popularity of an election or an advertising campaign [3], identify trends in political discourses [4], movie and product review mining [5], [6], and many more application areas.

Sentiment analysis (SA)(or opinion mining) focuses on discovering techniques that decipher these emotions and sentiments from raw text comments, reviews, etc. SA widely focuses on prediction or categorization of *polarity* encompassed within a text. The categorization could be into two categories such as *positive* or *negative* [7]; or even into a third *neutral* category as well [8]. Positive and negative text is sometimes also referred to as *subjective* text, whereas neutral text is referred to as *objective*. Similarly, SA also aims to predict *emotions* (happiness, sorrow, joy, anger, etc.) that are expressed within a text [9]–[11].

Quantifying the sentiments in text documents is not an easy task as they come from various domains, cover a wide variety of topics, and are often unorganized and unstructured. Predominantly two types of approaches exist for SA; supervised machine learning (ML) approaches, and the (unsupervised) sentiment lexicon (SL) based approaches. Supervised approaches are based on ML algorithms such as random forests, support vector machines, logistic regression, etc. They require a labelled set of text documents (referred to as training data) to learn the predictive model. On the other hand, lexicon-based approaches use pre-defined lexical dictionaries [12], thus not requiring labelled training examples. Lexicon based approaches can also be thought of as an expert system or a knowledge-based approach. Supervised approaches have the advantage of achieving higher accuracy, but their reliance on labelled data is a bottleneck as it requires a tedious process of reading through each text document and labelling it as positive or negative accordingly.

On the other hand, SL based approaches do not require labelled training data and thus can be applied directly without learning any training model. However, they suffer from a coverage problem, i.e., they fail to assign a sentiment label to each document. To reach the best of both worlds, we propose a hybrid approach that uses SL to transform the document term space into a two-dimensional sentiment feature space where an ML classifier is learned in a supervised fashion. This hybrid approach yields higher accuracy (or similar accuracy with fewer training examples) than the ML approaches, takes lesser time and memory to train than SL approaches. It is suitable for scenarios where training data is scarce. We support our claim by reporting results on six different online social media datasets (BBC, Digg, MySpace, Runner World, Twitter, and YouTube), using five SL (Afinn, Happiness index, SenticNet, SentiStrength, and SentiWordNet), and four ML algorithms (support vector machines, naïve Bayes, decision trees, and LogitBoost). Thus briefly, the main contribution of this are:

- Proposal and implementation of a hybrid technique for sentiment analysis.
- Study the effectiveness of the proposed approach, in comparison to baseline methods of pure machine learning and lexicon-based methods.

- Evaluation of proposed methodology with the different lexicon, machine learning algorithms and sentiment datasets.

The rest of the paper is organized as follows; literature review is given in Section II, followed by the description of our methodology in Section III. Dataset details, evaluation setup, and performance metrics are also described in this section. Section IV presents the results and the discussion about the performance of the various models. Finally, we conclude and state our future direction in Section V.

II. Related Work

The papers [11] and [13] provide detailed surveys of various ML and SL approaches used in the literature for sentiment analysis. For this study, we discuss here the most standard and widely used SL and ML approaches, followed by a detailed survey of hybrid approaches. We, therefore, structure the related work in three subsections according to these three types of approaches.

A. Lexicon-Based Techniques

All the lexicon-based approaches have in common a dictionary of words (or phrases) having some score that hints towards their polarity. They differ in terms of the source of these words, dictionary size, and the methodology used to assign them a score [1]. The process of building such a lexicon is subjective; therefore, all these dictionaries only have a small overlap. Similarly, a word may be deemed by one expert to have a positive sentiment, whereas others may deem it as neutral or even negative. Furthermore, many words inherently do not contain a positive or negative orientation, but it is the context in which they are used that makes their polarity positive or negative [14]. Due to the aforementioned reasons, there is no standard lexicon dictionary. Often there might be a tweet, or a blog, that contains no word that has a polarity, in which case, it is said that the lexicon does not cover that particular document and no score is assigned to it. This problem is referred to as the coverage problem [15]. Lexicon based approaches have a major benefit of not requiring any data for training, and thus can be used as off the shelf solutions. We chose the four lexicons provided by the *iFeel* utility [12], [16], namely, *SenticNet*, *SentiWordNet*, *Happiness Index*, and *SentiStrength*. The fifth lexicon used is *AFINN* [17]. These five approaches are described below.

1) Happiness Index: Happiness Index proposed by [18], calculates average psychological scores and frequency for the Affective Norms for English Words (ANEW) dictionary [19]. ANEW is a set of one thousand and thirty-four words bearing scores for psychological valence (good–bad), dominance (strong–weak), and arousal (active–passive) and their semantic differentials. Based on these dimensions, words are assigned a happiness score on the scale between 1 to 9. In our study, we consider the words with scores between 1 to 4 as negative words, whereas words with scores between 5 to 9 are regarded as positive words.

2) SentiStrength: SentiStrength is a hybrid approach that combines supervised and unsupervised classification methods [20]. It consists of two thousand three hundred and ten words exhibiting sentiments based on the Linguistic Inquiry and Word Count (LIWC) dictionary [21]. Each word has a human-assigned sentiment score from -5 to 5. Words having a score from -5 to -1 are considered as negative, whereas words having a score from 1 to 5 are regarded as positive words. SentiStrength divides the given documents into words and removes punctuations and emoticons, but in our study, we already remove these artefacts during the pre-processing phase. An associated sentiment score defined by SentiStrenght is then mapped to each word. The scores are then summed up for the positive and negative category. The category with the higher score is then marked as the category of that particular document.

3) SenticNet: SenticNet is a concept-level knowledge base that provides a set of sentics, semantics, and polarity for 100,000 concepts. It uses AI and semantic web techniques on web content to recognize, process, and interpret natural language opinions. SenticNet assigns a sentiment score to each concept between the range from -1 to 1 [22]. In this study, we consider words with values less than 0 as negative words, whereas words with scores higher than 0 are considered as positive words.

4) AFINN: AFINN [23] is a sentiment lexicon worked out by Finn Årup Nielsen group that is based on data generated by Twitter. It is based on 1000 tweets used in "Twitter mood maps reveal emotional states of America" [24]. This lexicon has a total of 2,477 words labelled within the integral range of ±[1...5], where the positive and negative signs indicate whether the word is positive or negative, respectively. The larger the score, the more intense, is the sentiment.

5) SentiWordNet: SentiWordNet [8] is based on popular English lexical dictionary 'WordNet' [25]. SentiWordNet has more than 117,000 words, including nouns, verbs, and adjectives. Each word is assigned a positive and as well as a negative score within the $[0 \ldots 1]$ range.

B. Machine-Learning Methods

Supervised ML aims to learn a predictive model from information encompassed in a given (training) dataset and then apply that model to another (testing) dataset for predictions. After document pre-processing, supervised learning is performed using a cross-validation approach, and yielded results are saved accordingly. [26] provides a detailed survey of supervised ML methods used for sentiment analysis. For the purpose of this study we choose three benchmark machine learning algorithms namely, $LogitBoost$ (LB), $naïve\ Bayes$ (NB), and support vector machines (SVM).

1) Naïve Bayes: NB is a probabilistic classifier that has been widely used for text classification in general and sentiment analysis in particular [27]. Based on Bayes theorem, it assumes a naïve independence i.e. all attributes are independent of each other given the category label. Even though this assumption does not hold in most cases, the resulting model is easier to fit and works remarkably well for large dimensional problems. NB predicts the class with the most probable hypothesis. Thus NB assigns the label $\hat{y} = c_k$ for the document X according to the following equation:

$$\hat{y} = \underset{k \in \{P,N\}}{\arg\max}\, p(C_k) \prod_{i=1}^{n} p(x_i|C_k) \tag{1}$$

Where x_i represents the i^{th} attribute (or feature) of document X, and P and N represent the positive and negative class labels, respectively.

2) Decision Trees: Decision trees are well known non-parametric supervised learning methods used for classification and regression. Although initially proposed more than three decades ago [28], newer versions are still popular today [29]. DT predicts the value of the target variable by inferring simple if-then-else type decision rules from the dataset. DT are visualized as a tree structure in which internal nodes perform a check on the attribute values, whereas the leaf nodes correspond to an outcome of the target attribute. For prediction, each record is traversed through the root node until it reaches the leaf node where it is assigned a prediction label that is associated with that particular leaf node. Attributes are selected using an information-theoretic measure called entropy. We use an advanced implementation of the DT algorithm named as classification and regression trees (CART) [30].

3) Support Vector Machines: SVM is a widely used discriminative classifiers for classification of text and sentiments [31], [32]. It is a binary classifier that projects each document as a point in a higher dimensional feature space such that the points belonging to the different categories are separated as far as possible. A hyperplane is then learned in this feature space to discriminate between the points of the two categories. The learned hyperplane is an optimal hyperplane i.e. it maximizes the gap (or distance or margin) between the closest points of the two categories. This help SVM achieve a better generalization over the unseen data. Decision function for SVM is as follows [33]:

$$\hat{y_i} = \begin{cases} P & \text{if } \mathbf{w^T} \cdot \mathbf{X}_i - b \geq 1 \\ N & \text{if } \mathbf{w^T} \cdot \mathbf{X}_i - b \leq -1 \end{cases} \tag{2}$$

Optimization of above objective function through maximizing marginal width and penalizing the vectors that fall within the hyper plane leads to the following decision function:

$$\arg\min_{\mathbf{w},\xi,b} \begin{cases} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \\ \sum_{i=1}^{n}\alpha_i[y_i(\mathbf{w^T}\cdot\mathbf{X_i} - b) - 1 + \xi_i] \end{cases} \tag{3}$$

Where $\mathbf{w}$ is defined as the weight vector; ξ_i as the error term i.e. vectors found within the marginal boundary of hyperplane; and $C > 0$ as the regularization parameter [33].

4) LogitBoost: LogitBoost is a boosting algorithm that has been shown to classify text documents successfully [34]. Boosting is an ensemble approach that combines many weak classifiers to come up with a strong or good classifier that primarily reduces bias and variance. It sequentially fits multiple weak classifiers in a way such that more weight is given to observations in the dataset that were misclassified by the previous classifiers in the sequence. In essence, the training data is re-weighted to produce multiple classifiers in sequence [35]. Like AdaBoost, LogitBoost also performs additive logistic regression using maximum Bernoulli likelihood as a criterion.

C. Hybrid Approaches

Zhang et al. [36] propose an entity level hybrid approach for sentiment analysis for Twitter data. SL is used to perform sentiment analysis on a set of pre-defined entities. The sentiments assigned to these entities are used to identify new tweets having a similar sentiment. These tweets are then used as an automatically labelled training data to train an SVM, whereas we transform the term space using the SL into a two-dimensional feature space instead of generating the labelled data. Furthermore, [36] approach requires pre-defined entities but no manually labelled training data, in contrast, we do not require the former but need the later. Additionally, we are not performing an entity-level sentiment analysis.

Martin et al. [37] combine SA and ML approaches in an ensemble setting. Separate ML classifiers are learned for film reviews in Spanish and its corresponding English translation. A third model is then learned by using the SL (SentiWordNet) on the translated English corpus. The decision of the three classifiers is combined using stacking or voting to output the final label. [38] also propose a hybrid ensemble method to infer sentiment from documents using statistical methods and knowledge-driven linguistic patterns. Our approach, on the other hand, does not ensemble SL and ML classifiers; rather, it uses the SL approach to transform the term space.

In [39], Prabowo et al. propose a hybrid approach that uses multiple SL, rule-based, statistical, and ML approaches in a cascading fashion. Their approach starts by predicting sentiment from one of the algorithms, and if it fails to assign a sentiment, the text is then passed to the next algorithm, and so on until it is eventually assigned a sentiment by one of the algorithms. They experiment with ten different sequences of general inquirer based classifier [40], statistics based classifier, decision tree classifier (ID3) [28], RIPPER [41], and SVM. [42] also propose a hybrid approach that cascades four classifiers: (a) an emoticon classifier, (b) a slang language classifier, (c) an improved domain-specific classifier, and (d) the SentiWordNet classifier. The input document is classified in a two-stage process. In the first stage, it is classified through the first two classifiers; then in the second stage, it is classified through the last two classifiers to get a more accurate classification. [43] also propose a hybrid approach consisting of a sequence of the following five components: (a) sentiment rules, (b) semantic lexicon, (c) ambiguity management process, (d) negation handling process, and (e) linguistic variables. Our approach, to the contrary, is not a cascading classifier approach; we only learn a single discriminative classifier. That is because the SL is used by us to serve to transform the term space only.

Wiebe et al. [44] propose a hybrid approach for classifying sentences into subjective and objective categories. First, lexicons of subjective and objective clues are used to label the data as subjective or objective. The patterns for each category are then extracted from this automatically labelled data. These patterns then serve as a new training data for NB classifier that is used to label the whole of the unlabeled data even that which were left out during the initial labelling by the lexicons. Whereas, our approach focuses on classifying objective texts into positive and negative classes; secondly, we use SL for feature transformation rather than sampling a portion of the unlabeled data for further feature extraction.

The closest approach is given by Ghorbel et al. [45], which shows a hybrid approach for classifying movie reviews in the French language. They translate the French words into English and then find their polarity score using the SentiWordNet sentiment lexicon, after employing some word disambiguation. The polarity score is then used in conjunction with the French text, its POS tags, and some other features to build a feature vector to train an SVM model. Our approach differs with their approach in the sense that we do not use the sentiment scores as features to complement the textual attributes instead we transform the term space into a two-dimensional feature space using these sentiment scores.

In [46] use a three-step hybrid approach. First, they project the data onto an SL and then augment it with a word embedding. This transformed input space is then served as input to ML algorithms. They use the unsupervised Word2Vec embeddings developed by researchers at Google. It exploits the co-occurrence of words in a corpus to detect the meanings and semantic relations between words by training a Deep Neural Network. These approaches have recently gained popularity for sentiment analysis [47]–[49]. Our approach differs from these techniques as we do not learn any embeddings using deep learning and nor we project the input terms using these embeddings rather we transform the document term space through SL into a two-dimensional sentiment space.

Similarly, Mudinas et al. [50] propose a hybrid approach that uses an SL to generate a feature vector for an ML algorithm (SVM). They use SL and POS tagging to generate a feature vector containing sentiment words, adjectives, and lexicon-based sentiment scores, which are then used to train an SVM model. We, on the other hand, do not project our data on to the term space of the SL; instead, we transform the document term space using SL into a two-dimensional sentiment space. Furthermore, they perform concept-level sentiment analysis with the aim of learning optimal weights for these concepts, whereas we are doing document-level sentiment analysis.

The approach closest to what we propose in this paper is given by [51], where hybrid approach uses different sentiment and emoticon lexicons to transform the document term space into seven feature vector space consisting of the frequencies of positive words, very positive words, negative words, very negative words, booster words, negation words, positive emotions, and negative emoticons. Like our approach, this feature vector is then used to train an SVM model. Our approach is different from [51] in the sense that we employ lexicons to find the polarities of each word which are then ensembled into two scores, one for the positive category and one for the negative category. Our discriminative model is learned on this transformed two-dimensional sentiment space.

III. Methodology

Sentiment classification is the prediction of a discrete-valued sentiment. It determines the sentiment of a textual document, whether it be a tweet, Facebook post, product review, SMS, etc. Therefore our target is to predict the overall sentiment of the textual document as either positive or negative. Hence, the problem formulated as such becomes a binary sentiment classification problem in which the class (or category) label P refers to documents exhibiting a positive sentiment, whereas class label N refers to documents where the negative sentiment is predominant. The class labels are just symbolic labels that do not carry any semantics or any additional knowledge. For this study, class P is treated as the positive class.

The binary sentiment classification problem of text documents is formally defined as follows. Let $L = \{\langle \mathbf{x}_i, c_i \rangle\}_{i=1}^{\|L\|}$ be a labeled set of documents such that $c_i \in C = \{P, N\}$ represents the sentiment of the i^{th} document $\mathbf{x}_i$ and $\|L\|$ is the total number of documents in L; learn a classification model that assigns a class label c_i to each document in the unlabeled set $U = \{\langle \mathbf{x}_i \rangle\}_{i=1}^{\|U\|}$. It is assumed (although not guaranteed in practice) that the joint probability distribution of the text documents and the target variable C is identical in the labeled (L) and unlabeled sets (U). Therefore, our task is to approximate the unknown target function $\Phi^{'} : U \rightarrow C$ by the classifier function $\Phi : U \rightarrow C$ such that the number of $x_j \in U$ for which $\Phi(\mathbf{x}_j) \neq \Phi^{'}(\mathbf{x}_j)$ is as less as possible. For lexicon-based approaches, the classifier function $\Phi : U \rightarrow \{P, N\}$ is directly derived from lexicon dictionary and hence there is no learning of classifier function $\Phi : U$ from the labeled set L. We represent the text document x_j as a integer valued vector $\mathbf{x}_i = \langle x_{i1}, x_{i2}, \ldots, x_{i\|A\|} \rangle$ in a $\|A\|$ dimensional vector space such that x_{ij} indicates the value of the feature j for the text document i and $\|A\|$ is the number of unique features in the set $L \cup U$ after standard pre-processing has been applied.

Our proposed approach combines the lexicon and learning-based approaches into a single hybrid approach. We describe its schema and steps in the subsequent subsection. Data and its cleaning process are also discussed in this section.

A. *Proposed Hybrid Approach*

The proposed approach consists of two major parts, the unsupervised feature transformation, and the supervised discriminant classification. The unsupervised part relies on SL to transform the $\|A\|$ dimensional term space into a two-dimensional sentiment vector space where each dimension corresponds to one of the sentiment class in the set C. Each dimension represents the opinion of the terms based on their respective sentiment polarity score in one of the lexicons. The terms with positive polarity contribute towards the membership of the document for positive sentiment class P, whereas the negative polarity terms contribute towards the membership of the document for the negative sentiment class N. The aggregated polarity score of all these terms is obtained as a linear sum of individual polarity scores normalized by the total number of words in the document. The resultant two scores S^{P} and S^{N} are thus the normalized sentiment scores for the positive and negative classes for document X, respectively. A term contributes towards the score only if it occurs in that document. If the same term occurs more than once in the document, then it contributes to the score S^{P} (or S^{N}) each time.

The sentiment scores, S^{P} and S^{N} define the two-dimensional sentiment space in which documents are aligned along the dimension that corresponds to the sentiment prevalent in them. In this space, documents belonging to one sentiment class are easily discriminable from the documents belonging to the other sentiment class, as illustrated for a dataset in Fig. 2. In this figure, the documents having positive sentiment as the true label (coloured in purple) nicely

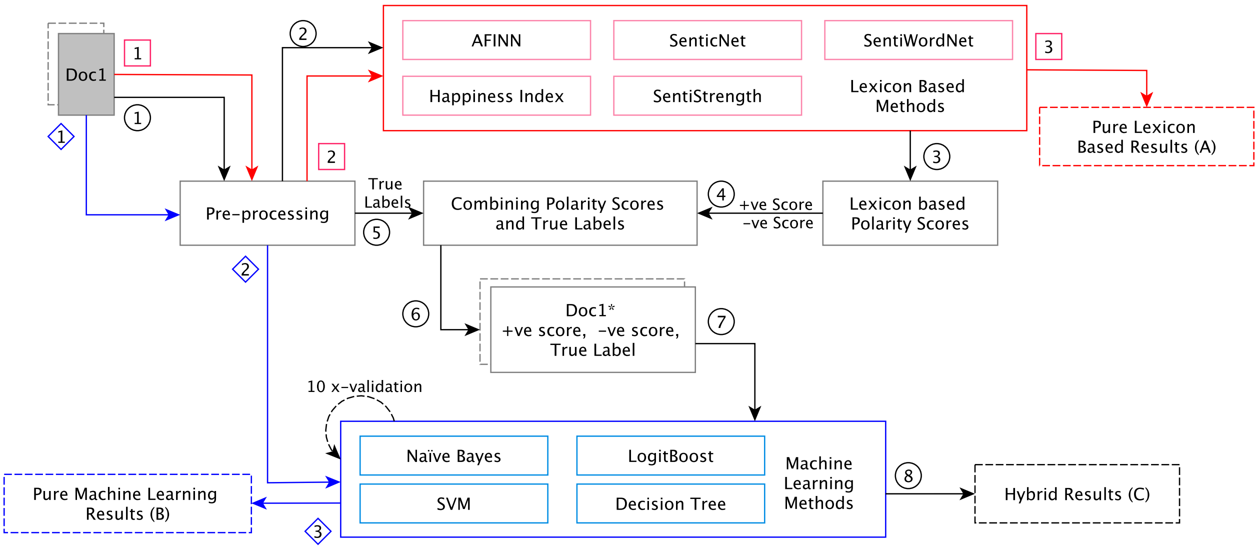

Fig. 1. Block Diagram of the Overall Process. The Arrows in Red Color (Tagged with in Squares) Correspond to the Sentiment Lexicon Approaches, whereas Blue Colored Arrows (Tagged with in Diamonds) Correspond to Pure Machine Learning Approaches, and Arrows in Black Color (Tagged with in Circles) Correspond to the Proposed Hybrid Approach.

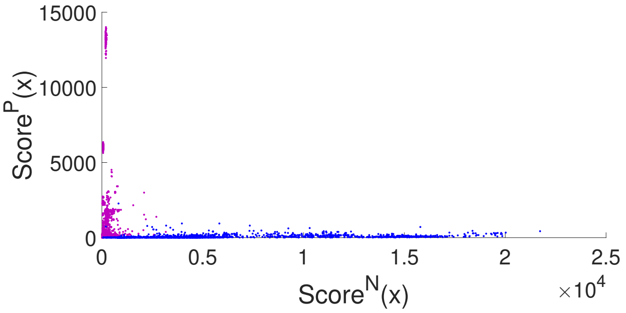

Fig. 2. The 2D Feature Space after Applying SL Approach. Each Point in this Sentiment Space Corresponds to a Document in the Dataset. Purple and blue points are documents with P and N as true class label, respectively.

align along the y-axis whereas the documents having negative sentiment as the true label (coloured in blue) align along the x-axis. Thus these sentiment scores can be thought of as the confidence values of a document's membership in the positive and negative class.

There are multiple hypotheses possible in this two-dimensional space. A potential hypothesis could be to assign the highest sentiment score class to the document. This is also known as the max rule and corresponds to a straight line at a 45 degrees angle from the origin. Though quite handy and intuitive, these rules fail to achieve good generalizations when there exists a class imbalance in the data set or the distribution of the training and test is significantly different. A better hypothesis may be a maximum margin hypothesis such as learned by SVM, or the hypotheses of some other discriminative classifier. Therefore, we train different supervised discriminative classifiers in this sentiment space to find the best hypotheses that separate the two sentiment classes. Fig. 1 depicts the overall schema of our methodology.

To generate the aforementioned two-dimensional sentiment space, we alter the decision rule of the SL approaches. Instead of providing a decision about whether the document is positive or negative, we make them output the positive or negative scores only. Algorithm 1 depicts the altered algorithm for SenticNet that outputs the S^{P} and S^{N} scores instead of outputting the label. The true labels of the document are appended to these two scores to obtain the training data for the supervised approach. Thus the dimensionality of our problem is reduced

```
Data: Let $\mathbf{X}$ be the text to process; $\mathbf{D}$ be the
      SenticNet dictionary
Result: A tuple of information ⟨coverage, positive
        score, and negative score⟩
c ← be a variable for coverage intially set to False
$S^{\mathrm{P}}$ ← be a variable for positive score initially set to 0
$S^{\mathrm{N}}$ ← be a variable for negative score intially set to 0
Remove all punctuation marks in $\mathbf{X}$
Remove all stopwords in $\mathbf{X}$
if $\| \mathbf{X} \| = 0$ then
    return ⟨c, $S^{\mathrm{P}}$, $S^{\mathrm{N}}$⟩
end
for each token $w$ in $\mathbf{X}$ do
    if $w \in \mathbf{D}$ then
        c ← True
        if $\mathbf{D}[w] > 0.0$ then
            $S^{\mathrm{P}} \leftarrow S^{\mathrm{P}} + \mathbf{D}[w]$
        else
            $S^{\mathrm{N}} \leftarrow S^{\mathrm{N}} + \mathbf{D}[w]$
        end
    else
        continue
    end
end
return ⟨c, $S^{\mathrm{P}}$, $S^{\mathrm{N}}$⟩
```

Algorithm 1: A Sample of Altered Algorithm (SenticNet) for Proposed Hybrid Approach.

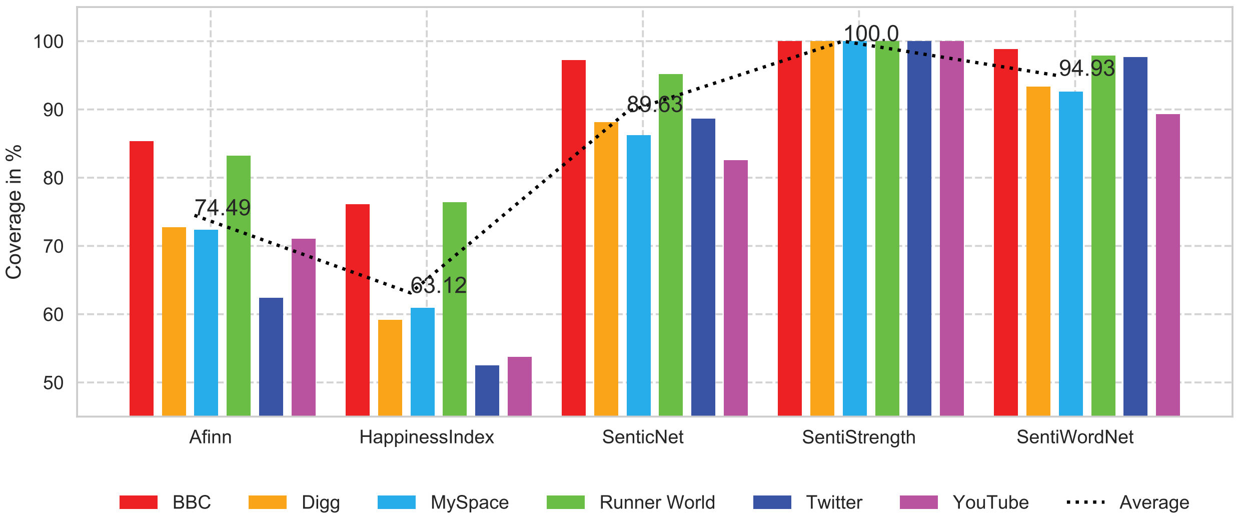

Fig. 3. Coverage of Lexicon based Methods. Each Number Indicates the Percentage of Records Covered in Dataset.

from the number of unique terms ($\|A\|$) in the dataset to just a two-dimensional sentiment space. The resultant feature space is more comfortable to visualize and gives insights on the separability of the sentiments in the transformed feature space. It is also faster for the ML algorithms to build their model because of the small feature space. It is to be noted that our hybrid approach is not specific to any SL or ML algorithm. Any SL can be used to generate the two-dimensional sentiment space over which any ML algorithm can be applied to learn the final decision function.

TABLE I. STATISTICS OF DATASETS: NUMBER OF RECORDS (DOCUMENTS) IN IT AND PERCENTAGE OF CLASS DISTRIBUTION.

Dataset	# Records	% of +ve / -ve Records
BBC	1,000	34.70% / 65.30%
Digg	1,077	46.89% / 53.11%
MySpace	1,041	87.32% / 12.68%
Runner World Forum	1,046	78.87% / 21.13%
Twitter	4,242	77.63% / 22.37%
YouTube	3,407	77.49% / 22.51%

B. Pre-Processing and True Label Generation

We chose the datasets used by [20] to evaluate the performance of the classifiers. The dataset belongs to six online domains, namely, BBC, Digg, MySpace, Runner World, Twitter, and YouTube. Details of these datasets are given in Table I. As given in Table I, the datasets have different sizes and have a very different distribution of positive and negative documents. Each record in these datasets is labeled (scored) by humans using Mechanical Turk [52] based on positive and negative sentiment present in them. Thus, each record has it is a positive score and as well as a negative score. The following rule is used to assign a class label to each document X:

$$\mathrm{T}(X_i) = \begin{cases} Positive & \text{if +ve score}_{X_i} \geq \text{+ve score}_{X_i} \\ Negative & \text{otherwise} \end{cases} \tag{4}$$

Where i is the document index $1 \geq i \leq n$.

The following preprocessing steps are applied to the data before running any of the classifiers.

1) Punctuation marks are removed by replacing them by empty string using the following regular expression (R) `[\@\#\$\%\^\&\*\_\.?\!\:\,\;\+\=\-\|\<\>\{\}\(\)\[\]\"\/]`.
2) Case folding is performed to transform whole document to lower case.
3) Documents are transformed into a term-incidence matrix (for ML approaches only).

In addition to the above two steps, SL methods also have their data cleaning steps. E.g., SenticNet replaces characters `.!?,` with a single whitespace, followed by tokenization of text to be processed with simple white space (line 2, Algorithm 1).

IV. RESULTS AND EVALUATIONS

Before we can compare SL approaches with ML and the proposed hybrid (or combined) approach, it is important to note that SL approaches do not assign a label to every example in a dataset, a problem known as the coverage problem. It occurs when no word of the example to be classified is present in the SL. For such examples, the SL cannot give any decision regarding their sentiment. The coverage of the five sentiment lexicons used in this study is given in Fig. 3. SentiWordNet with more than 117,000 words and phrases in its dictionary has the second-highest coverage $\approx$ 95%, while Happiness Index with 1,032 words only, attained the lowest coverage of 63.12%. However, SentiStrength demonstrates full coverage, which is not a surprise as it was created from the five datasets that are used in this study. Therefore, in order to compare the performance of the SL approaches to ML, and the proposed hybrid approach, it is necessary to have them tested on the same set of examples. Therefore, if a particular SL covers only 600 examples out of the 1000 examples, then the ML and the proposed approach classifiers are trained over these 600 examples only using a ten-fold cross-validation approach.

TABLE II. Performance comparison of all the approaches dataset wise. All values are in percentages.

Approach	Dataset	Precision	Recall	Specificity	F-Measure	Accuracy
Sentiment Lexicons	BBC	0.3903	0.7547	0.2271	0.5039	0.5055
	Digg	0.4941	0.7634	0.3239	0.5904	0.5360
	MySpace	0.8987	0.8735	0.6733	0.8848	0.8048
	Runner World	0.8132	0.8534	0.6335	0.8304	0.7296
	Twitter	0.8035	0.8307	0.5830	0.8147	0.7155
	YouTube	0.8011	0.8480	0.5740	0.8273	0.7300
Avg. Sentiment Lexicon		0.7018	0.8206	0.5025	0.7419	0.6702
Pure Machine Learning	BBC	0.4411	0.1837	0.8694	0.7038	0.6305
	Digg	0.5802	0.4450	0.7172	0.6603	0.5923
	MySpace	0.8881	0.9733	0.1231	0.7712	0.8638
	Runner World	0.8043	0.9400	0.1300	0.7479	0.7710
	Twitter	0.8133	0.9329	0.1903	0.7566	0.7692
	YouTube	0.8242	0.9161	0.2415	0.7504	0.7635
Avg. Pure Machine Learning		0.7252	0.7318	0.3785	0.7317	0.7317
Proposed Hybrid Approach	BBC	0.3838	0.1553	0.9337	0.2000	0.6790
	Digg	0.4654	0.2655	0.8820	0.2993	0.6123
	MySpace	0.8839	0.9852	0.0948	0.9313	0.8731
	Runner World	0.8000	0.9797	0.0638	0.8799	0.7886
	Twitter	0.7882	0.9511	0.1686	0.8599	0.7671
	YouTube	0.7911	0.9652	0.1434	0.8678	0.7746
Avg. Proposed Hybrid Approach		0.6854	0.7170	0.3810	0.6730	0.7491

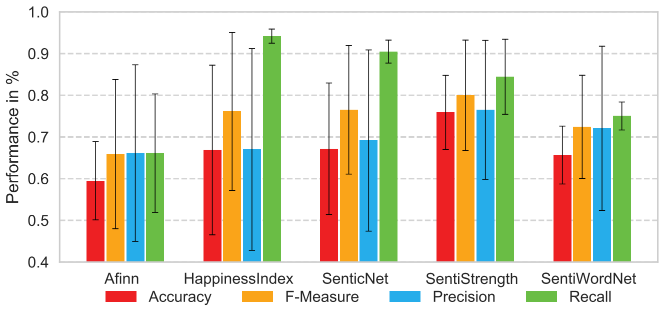

Fig. 4. Average Performance and Standard Deviation of Lexicon-based Approaches.

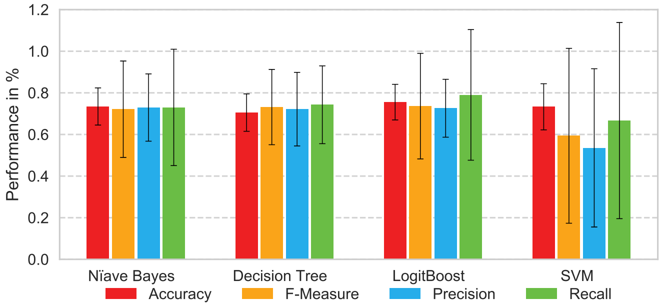

Fig. 5. Average Performance and Standard Deviation of (Pure) Machine Learning Approaches.

Therefore all the results for ML and the hybrid approaches are based on only those comments that are covered by that particular lexicon. The detailed comparative results for the SL, ML, and proposed hybrid classifiers is presented in Fig. 4, 5, and 6, respectively. It is to be noted at each value bars in these figures correspond to the values of the corresponding approach averaged over the six datasets mentioned in III-B.

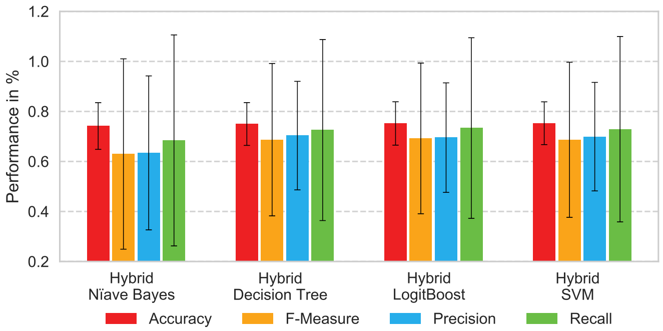

Fig. 6. Average Performance and Standard Deviation of the Proposed Hybrid Approach.

Not surprisingly SentiStrength outperforms the rest of the SL approaches (Fig. 4) because its dictionary was built upon the datasets used in this research work. Therefore, SenticNet seems to be the winner (after excluding SentiStrength) among SL approaches with the highest accuracy and F-measure, but it has a high standard deviation. Happiness Index is not far behind SentiStrength and has a higher recall but at the expense of lower precision. This can be attributed to a bias towards the positive class; furthermore, it has even more variance than SentiStrength. Overall the performance of the SL is not encouraging as the accuracy is below than 70% even when only those comments are used which are covered by these lexicons.

As expected, ML-based approaches perform better than SL approaches because they have a training phase (Fig. 5). All the four classifiers achieve an accuracy of more than 70% with LogitBoost outperforming the rest with more than 75% accuracy. This is a bit surprising as NB and SVM are thought as better classifiers for text classifiers. It is to be noted that each value bar in this figure corresponds to the average performance of a classifier over all the datasets using only the examples that are covered by the respective SL. E.g., the NB's precision of

71.01% is calculated from the examples of the six datasets that were covered by Afinn, and from the examples of six datasets that were covered by SenticNet, and so. Therefore each value in this table is an average of the performance of thirty different NB classifiers. Surprisingly LogitBoost outperforms NB and SVM.

As hypothesized, the proposed hybrid classifiers outperform the SL and ML classifiers (Fig. 6). The accuracy of all the four SL approaches increases when used with the proposed approach with decision tree benefiting the most with an average accuracy improvement of more than 4%. The comparison of all the approaches on individual datasets is presented in Table II. Performance gain over the SL approach was expected as SL is an unsupervised approach, whereas the ML and the proposed approach is a supervised one. The dataset to benefit the most from the proposed approach is the BBC dataset with an increase of 17.25% and 4.85% in average accuracy over the SL and ML approaches, respectively. For the Twitter data only does the ML approach beat the proposed hybrid approach but by only a margin of 0.21%. Overall the proposed approach outperforms the SL and ML approaches by a margin of 7.88% and 1.70%, respectively. Average accuracy improvement of 1.70% in the accuracy of the hybrid approach is significant, considering that the average accuracy of the ML approaches is about 73% only. The detailed performance gains are reported in Tables III and IV, respectively. SVM seems to be the major benefactor of the proposed approach.

TABLE III. Average improvement in percentage accuracy by the proposed hybrid approach over SL approaches. HI, SN, SS, SWN, and LB refer to the Happiness Index, SenticNet, SentiStrength, SentiWordNet, and LogitBoost, respectively.

	H-NB	H-DT	H-LB	H-SVM	Average
Afinn	14.68%	15.50%	15.71%	15.81%	15.42%
HI	7.28%	8.10%	8.31%	8.41%	8.02%
SN	6.98%	7.80%	8.01%	8.11%	7.72%
SS	-1.75%	-0.93%	-0.72%	-0.62%	-1.01%
SWN	8.49%	9.31%	9.52%	9.62%	9.23%
Average	7.13%	7.95%	8.17%	8.26%	7.88%

TABLE IV. Average improvement in percentage accuracy by the proposed hybrid approach over their ML counterparts.

	H-NB	H-DT	H-LB	H-SVM	Average
NB	0.71%	1.53%	1.75%	1.84%	1.46%
DT	3.64%	4.46%	4.68%	4.77%	4.38%
LB	-1.39%	-0.57%	-0.35%	-0.26%	-0.64%
SVM	0.87%	1.69%	1.90%	2.00%	1.61%
Average	0.96%	1.77%	1.99%	2.09%	1.70%

A. *Scalability and Complexity Analysis*

In terms of time and space requirements, the proposed approach is highly efficient than directly learning a supervised ML classifier. The two-dimensional sentiment score space is generated in a single pass over the labelled data. The time required to generate this space is $O(\|L\| \cdot a)$, where $\|L\|$ is the total number of labelled documents in the training data, $\|A\|$ as defined earlier is the number of unique terms in the dataset, and a is the average number of terms in a document. Since a document vector is sparse in the $\|A\|$ dimensional space, therefore, $a \ll \|A\|$, thus making the $O(\|L\| \cdot a)$ asymptotic running time linear in terms of the size of the dataset. The next major step on which the running time is dependent is the training of the supervised ML model is this two-dimensional sentiment score space. Thus the total time to generate the proposed model is $O((\|L\| \cdot a) + MTT)$, where MTT is the time taken to train the ML model. Depending on which ML model is used, MTT can also be linear in terms of the size of the dataset. Therefore the proposed algorithm is trained in linear time, and it is the fastest (asymptotically) running time for a binary class classification algorithm.

The proposed approach is more scalable than an approach that learns the ML algorithm directly. It is because the first step in generating the two-dimensional sentiment space requires hash table lookups to retrieve the score of a word and sum them up. A hash table lookup requires $O(1)$ time as the size of the SL is already known and is static. Furthermore, the size of the hash table of SL is much lesser than $\|A\|$. In the second step, the ML model is learned in a space with two attributes only which are very efficient as compared to when the ML model is learned directly in a $\|A\|$ dimensional term space because $\|A\|$ can easily reach hundreds of thousands of words.

Classification model of the proposed approach requires lesser space than the ML approaches by order of magnitude. E.g., NB calculates probabilities of each of the $\|A\|$ words for each class C, thus making it space complexity as $O(\|A\| \cdot \|C\|)$. Whereas for the proposed approach, in addition to the SL whose size is significantly lesser and as well as independent of the size of the data, only $O(\|C\|)$ probabilities are stored because of $\|A\|$ is equal to two in the two-dimensional sentiment score space.

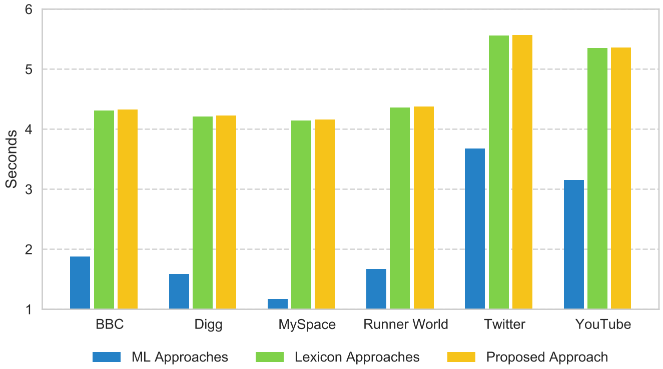

Fig. 7. Average Running Time of Algorithms on the Datasets.

Being a lazy learning approach, the SL approaches have the advantage of not having a training phase. However, the prediction of labels for the unseen data can be a bit slow as a dictionary lookup is required for each word in the document. Fig. 7 plots the average running times of the SL, ML, and the proposed approach to predict the labels for the various datasets. Since our approach uses the SL approach at the first step, its prediction is therefore deemed to be slower than the SL approach. However, the figure suggests that the overhead is very low as there is almost an overlap between the running time curves of the SL and the proposed approach.

Document representation as described in Section III corresponds to a sparse vector in the $\|A\|$ dimensional vector space. Using this representation, the whole data is represented as a document incidence matrix having a size of $\|L\| \cdot \|A\|$ dimensions. Since $\|A\|$ is a large number for text classification problems, therefore building such a big matrix requires much memory and computational cost. Approaches like NB do not require a document incidence matrix for computing its probabilities; instead, NB is efficiently implemented using a hash table data structure that would only require $O(\|L\| \cdot a)$ space. Since a is significantly less than $\|A\|$, it is a big improvement and makes NB one of the fastest classifiers. Like NB, the proposed approach is implemented using the hash table data structure, thus having a low memory footprint. Since we do not need to access terms and their scores in any specific order, therefore we retrieve, store and update the scores of each term in constant time using a hash table. This makes our approach very fast and memory efficient.

V. Conclusion and Future Work

Sentiment mining of textual content on social media can give insights for targeted advertisement, product reviews, and much more. In this article, we have proposed a hybrid sentiment analysis technique that uses sentiment lexicons (SL) to transform the input term space into a sentiment score space of only two dimensions where a supervised machine learning (ML) algorithm is learned to output the final decisions. The proposed approach demonstrates significant performance gain over the original SL and ML approaches when evaluated for three ML algorithms and five SL over six social media datasets. It also takes less time and memory to train than the ML approaches. Thus our approach is suitable for scenarios where training data is scarce, and more balanced classification is required. In the future, we plan to identify the terms in the SL that result in the classification bias & devise a mechanism to penalize them for reducing the bias of the proposed hybrid approach further.

References

[1] A. Jurek, M. D. Mulvenna, and Y. Bi, "Improved lexicon-based sentiment analysis for social media analytics," *Security Informatics*, vol. 4, no. 1, p. 9, 2015.

[2] J.-Y. Chen, H.-T. Zheng, Y. Jiang, S.-T. Xia, and C.-Z. Zhao, "A probabilistic model for semantic advertising," *Knowledge and Information Systems*, pp. 1–26, 2018.

[3] M. M. Mostafa, "More than words: Social networks' text mining for consumer brand sentiments," *Expert Systems with Applications*, vol. 40, no. 10, pp. 4241–4251, 2013.

[4] D. Liu and L. Lei, "The appeal to political sentiment: An analysis of donald trump's and hillary clinton's speech themes and discourse strategies in the 2016 us presidential election," *Discourse, Context & Media*, 2018.

[5] W. Wang, H. Wang, and Y. Song, "Ranking product aspects through sentiment analysis of online reviews," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 29, no. 2, pp. 227–246, 2017.

[6] W. Muhammad, M. Mushtaq, K. N. Junejo, and M. Y. Khan, "Sentiment analysis of product reviews in the absence of labelled data using supervised learning approaches," *Malaysian Journal of Computer Science*, vol. 33, no. 2, pp. 118–132, 2020.

[7] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002, pp. 79–86.

[8] A. Esuli and F. Sebastiani, "Sentiwordnet: A publicly available lexical resource for opinion mining," in *Proceedings of LREC*, vol. 6. Citeseer, 2006, pp. 417–422.

[9] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "Recognition of affect, judgment, and appreciation in text," in *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 806–814.

[10] S. Mohammad, "From once upon a time to happily ever after: Tracking emotions in novels and fairy tales," in *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Association for Computational Linguistics, 2011, pp. 105–114.

[11] K. Sailunaz, M. Dhaliwal, J. Rokne, and R. Alhajj, "Emotion detection from text & speech: a survey," *Social Network Analysis & Mining*, vol. 8, no. 1, p. 28, 2018.

[12] P. Gonçalves, M. Araújo, F. Benevenuto, and M. Cha, "Comparing and combining sentiment analysis methods," in *Proceedings of the first ACM conference on Online social networks*. ACM, 2013, pp. 27–38.

[13] Z. Hailong, G. Wenyan, and J. Bo, "Machine learning and lexicon based methods for sentiment classification: A survey," in *Web Information System and Application Conference (WISA), 2014 11th*. IEEE, 2014, pp. 262–265.

[14] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[15] J. Eisenstein, "Unsupervised learning for lexicon-based classification." in *AAAI*, 2017, pp. 3188–3194.

[16] M. Araújo, P. Gonçalves, M. Cha, and F. Benevenuto, "ifeel: A system that compares and combines sentiment analysis methods," in *23rd international conference on World wide web companion*. International World Wide Web Conferences Steering Committee, 2014, pp. 75–78.

[17] F. Å. Nielsen, "A new anew: Evaluation of a word list for sentiment analysis in microblogs," *arXiv preprint arXiv:1103.2903*, 2011.

[18] P. S. Dodds and C. M. Danforth, "Measuring the happiness of large-scale written expression: Songs, blogs, and presidents," *Journal of Happiness Studies*, vol. 11, no. 4, pp. 441–456, 2010.

[19] M. M. Bradley and P. J. Lang, "Affective norms for english words (anew): Instruction manual and affective ratings," Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, Tech. Rep., 1999.

[20] M. Thelwall, "Heart and soul: Sentiment strength detection in the social web with sentistrength," *Proceedings of the CyberEmotions*, pp. 1–14, 2013.

[21] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: Liwc and computerized text analysis methods," *Journal of language and social psychology*, vol. 29, no. 1, pp. 24–54, 2010.

[22] E. Cambria, C. Havasi, and A. Hussain, "Senticnet 2: A semantic & affective resource for opinion mining & sentiment analysis." in *FLAIRS*, 2012, pp. 202–207.

[23] F. Å. Nielsen, "A new anew: Evaluation of a word list for sentiment analysis in microblogs," *arXiv preprint arXiv:1103.2903*, 2011.

[24] C. Biever, "Twitter mood maps reveal emotional states of america," *New Scientist*, vol. 207, no. 2771, p. 14, 2010.

[25] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[26] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.

[27] J. Song, K. T. Kim, B. Lee, S. Kim, and H. Y. Youn, "A novel classification approach based on naïve bayes for twitter sentiment analysis," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 6, pp. 2996–3011, 2017.

[28] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[29] A. Ortigosa, J. M. Martín, and R. M. Carro, "Sentiment analysis in facebook and its application to e-learning," *Computers in Human Behavior*, vol. 31, pp. 527–541, 2014.

[30] R. A. Berk, "Classification and regression trees (cart)," in *Statistical learning from a regression perspective*. Springer, 2016, pp. 129–186.

[31] A. S. Manek, P. D. Shenoy, M. C. Mohan, and K. Venugopal, "Aspect term extraction for sentiment analysis in large movie reviews using gini index feature selection method and svm classifier," *World wide web*, vol. 20, no. 2, pp. 135–154, 2017.

[32] Y. Liu, J.-W. Bi, and Z.-P. Fan, "A method for multi-class sentiment classification based on an improved one-vs-one (ovo) strategy and the support vector machine (svm) algorithm," *Information Sciences*, vol. 394, pp. 38–52, 2017.

[33] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[34] S. Kotsiantis, E. Athanasopoulou, and P. Pintelas, "Logitboost of multinomial bayesian classifier for text classification," *International Review on Computers and Software (IRECOS)*, vol. 1, no. 3, pp. 243–500, 2006.

[35] J. Friedman, T. Hastie, R. Tibshirani *et al.*, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *The annals of statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[36] L. Zhang, R. Ghosh, M. Dekhil, M. Hsu, and B. Liu, "Combining lexicon based and learning-based methods for twitter sentiment analysis," *HP Laboratories, Technical Report HPL-2011*, vol. 89, 2011.

[37] M.-T. Martín-Valdivia, E. Martínez-Cámara, J.-M. Perea-Ortega, and L. A. Ureña-López, "Sentiment polarity detection in spanish reviews combining supervised and unsupervised approaches," *Expert Systems with Applications*, vol. 40, no. 10, pp. 3934–3942, 2013.

[38] E. Cambria and A. Hussain, *Sentic computing: a common-sense-based framework for concept-level sentiment analysis*. Springer, 2015, vol. 1.

[39] R. Prabowo and M. Thelwall, "Sentiment analysis: A combined approach," *Jrn. of Informetrics*, vol. 3, no. 2, pp. 143–157, 2009.

[40] P. J. Stone, D. C. Dunphy, and M. S. Smith, "The general inquirer: A computer approach to content analysis." 1966.

[41] W. W. Cohen, "Fast effective rule induction," in *Proceedings of the twelfth international conference on machine learning*, 1995, pp. 115–123.

[42] M. Z. Asghar, F. M. Kundi, S. Ahmad, A. Khan, and F. Khan, "T-saf: Twitter sentiment analysis framework using a hybrid classification scheme," *Expert Systems*, vol. 35, no. 1, p. e12233, 2018.

[43] O. Appel, F. Chiclana, J. Carter, and H. Fujita, "Successes and challenges in developing a hybrid approach to sentiment analysis," *Applied Intelligence*, vol. 48, no. 5, pp. 1176–1188, 2018.

[44] J. Wiebe and E. Riloff, "Creating subjective and objective sentence classifiers from unannotated texts," in *Computational Linguistics and Intelligent Text Processing*. Springer, 2005, pp. 486–497.

[45] H. Ghorbel and D. Jacot, "Sentiment analysis of french movie reviews," in *Advances in Distributed Agent-Based Retrieval Tools*. Springer, 2011, pp. 97–108.

[46] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas, "Sentiment analysis leveraging emotions and word embeddings," *Expert Systems with Applications*, vol. 69, pp. 214–224, 2017.

[47] S. Xiong, H. Lv, W. Zhao, and D. Ji, "Towards twitter sentiment classification by multi-level sentiment-enriched word embeddings," *Neurocomputing*, vol. 275, pp. 2459–2466, 2018.

[48] E. Cambria, S. Poria, D. Hazarika, and K. Kwok, "Senticnet 5: discovering conceptual primitives for sentiment analysis by means of context embeddings," in *AAAI*, 2018.

[49] X. Fu, X. Sun, H. Wu, L. Cui, and J. Z. Huang, "Weakly supervised topic sentiment joint model with word embeddings," *Knowledge-Based Systems*, vol. 147, pp. 43–54, 2018.

[50] A. Mudinas, D. Zhang, and M. Levene, "Combining lexicon and learning based approaches for concept-level sentiment analysis," in *Proceedings of the First Int. Workshop on Issues of Sentiment Discovery and Opinion Mining*. ACM, 2012, p. 5.

[51] D. Mumtaz and B. Ahuja, "A lexical and machine learning-based hybrid system for sentiment analysis," in *Innovations in Computational Intelligence*. Springer, 2018, pp. 165–175.

[52] M. Buhrmester, T. Kwang, and S. D. Gosling, "Amazon's mechanical turk a new source of inexpensive, yet high-quality, data?" *Perspectives on psychological science*, vol. 6, no. 1, pp. 3–5, 2011.