A Lexical and Machine Learning-Based Hybrid System for Sentiment Analysis

Deebha Mumtaz and Bindiya Ahuja

Abstract Micro-blogs, blogs, review sites, social networking site provide a lot of review data, which forms the base for opinion mining or sentiment analysis. Opinion mining is a branch of natural language processing for extracting opinions or sentiments of users on a particular subject, object, or product from data available online Bo and Lee (Found Trends Inf Retrieval 2(1–2):1–135, 2008, [1]). This paper combines lexical-based and machine learning-based approaches. The hybrid architecture has higher accuracy than the pure lexical method and provides more structure and increased redundancy than machine learning approach.

Keywords Lexical analysis · Machine learning · Sentiment · Hybrid approach

1 Introduction

A massive quantity of data is generated daily by social networks, blogs, and other media on the World Wide Web. This enormous data contains vital opinion-related information that can be exploited to profit businesses and other scientific and commercial industries. However, manual mining and extraction of this information are not possible, and thus, sentiment analysis is required. Sentiment analysis is an application of natural language processing and text analytics for extracting sentiments or opinions from reviews expressed by users on a particular subject, area, or product online [1]. Its main aim is to classify each sentence or document as positive, negative, or neutral. The techniques employed so far can be categorized into lexical-based approach and machine learning approach.

The lexical approach is based on the supposition that the combined polarity of sentences is the summation of individual word polarity [2]. This approach classically employs a lexical or dictionary of words which are pre-labeled according to their polarity. Then, every word of the given document is compared with the words

D. Mumtaz (✉) · B. Ahuja
MRIU, Faridabad, Haryana, India
e-mail: deebhamumtaz@gmail.com

B. Panda et al. (eds.), *Innovations in Computational Intelligence*, Studies in Computational Intelligence 713, https://doi.org/10.1007/978-981-10-4555-4_11

in the dictionary [3]. In case the word is found, then its polarity strength value is added to the total polarity score of the text, and hence, the orientation is finally obtained. The main benefit of the lexical approach is that it is a simplistic method, fast, and works well for precise, small dataset. However, its disadvantages are low accuracy, low precision, low recall, and low performance in complex datasets.

In machine learning technique, a group of feature vector is selected and a labeled dataset, i.e., training data, is made available for training the classifier. The classifier can then be used to classify the untagged corpus (i.e., test data set). The choice of features is critical for the performance of the classifier [4]. Usually, a range of unigrams (single words) or *n*-grams (two or more words) is selected as feature vectors. Apart from these, the features may comprise of the frequency of opinion words, the frequency of negation words, strength of words, and the length of the text. Support vector machines (SVMs), max entropy, and the Naive Bayes algorithm are the most important classification algorithms used [5].

A. Naive Bayes classifier is based on Bayes' theorem, which assumes that the value of a particular feature is not dependent on the value of any other feature, i.e., each feature is independent. This model is easy to build, simple, and helpful for very large datasets, with no complicated iterative parameter estimation present [6, 7].
B. Maximum entropy is a technique for obtaining probability distributions of given data. The basic principle is that when no information is known, then the distribution should have maximal entropy [6]. Labeled training data provides constraints on the distribution, determining where to have the minimal nonuniformity [7].
C. Semi-supervised learning is a class between unsupervised learning and supervised learning that makes use of unlabeled data. Many researchers have found that unlabeled data, when used in combination with a small quantity of labeled data, can produce a significant enhancement of learning accuracy [7, 8].
D. Support vector machines are supervised techniques coupled with learning algorithms that examine data used for classification. Given a set of training examples, each distinctly labeled for belonging to one of the types. An SVM training algorithm builds a model that allot novel examples into one category or the other, making it a binary linear classifier which is non-probabilistic [3].

The advantage of machine learning approach is its high accuracy and high precision, and it works well on complex datasets. But the disadvantage is that the final output quality depends on the quality of training dataset, and it is less structured, sensitive to writing style, and has low performance in cross-style setting [9].

The proposed approach is based on combining the two techniques. Compared to lexical system, it achieves considerably high accuracy, and with respect to machine learning approach, it offers more structure, readability, and reliability. The remaining paper is organized as follows. In Sect. 2, a review of the related work is discussed. In Sect. 3, the proposed system based on hybrid approach is introduced. In Sect. 4, the experimental results are given and conclusion is presented in Sect. 5.

2 Related Work

Pang and Lee [1] assessed and compared the various supervised machine learning algorithms for classifying the opinions of movie reviews. They used learning algorithms such as Naive Bayes (NB), maximum entropy (ME), and support vector machine (SVM). With a straightforward algorithm using SVM, they trained on bag-of-words attributes and obtained 82.9% accuracy, which was later enhanced in their future work [10] to 87.2%. However, other researchers have found [11] that such a simple design experiences challenges of domain, time, and style dependencies. Additionally, it gives just a general sentiment value for each sentence exclusive of any clarification as to what people exactly loved or hated and up to what extent they did so.

Hu and Liu [12] put forward a bi-step process for sentence-level opinion mining, which was then enhanced by Popescu and Etzioni [12]. According to this process, opinion mining could be divided into two stages: aspect categorization and measurement of sentiment strength for each feature. Feature selection is a vital step since the aspects set up the domain in which the sentiment was expressed. Hu and Liu [12] found that aspect identification can be carried on by choosing the most commonly used nouns and noun phrases. But reviews from customers are frequently short, casual, and not precise, which makes this job very difficult. In order to overcome this challenge, labeled sequential rules (LSR) can be used [13], where such a rule is fundamentally a unique kind of sequential pattern.

Usually, most sentiment analysis techniques are based on either pure lexical or learning methods. However, with growing recognition of numerous generative probabilistic models, based on latent Dirichlet allocation (LDA), some attempts have been made to integrate lexical method into machine learning system [14, 15]. Davidov et al. [16] researched on the presence of hashtags and emoticons in opinion mining. Agarwal et al. [17] studied novel features for sentiment analysis of Tweets. Mittal et al. [18] proposed a hybrid system combining Naïve Bayes and lexical method and acquired an accuracy of 73%. They also incorporated tools for spelling correction (e.g., gooood as good) and slang handling. Malandrakis [19] combined lexical and max entropy algorithm for sentiment analysis with POS tagging, emoticons, and *n*-grams. Harsh [20] combined SVM and Naive Bayes algorithm and acquired an accuracy of 80%.

3 Proposed Approach

The proposed system is a combination of both lexical and machine learning techniques, hence exploiting the best characteristics of both in one. For this purpose, we choose to utilize a support vector machine (SVM) and empower it with an English lexical dictionary AFINN [8]. Our hybrid system follows the conventional four steps, namely data collection, preprocessing, training the classifier, and classification.

Through the following sections, we shall discuss each step in detail, one at a time. Figure 1 shows the system architecture of the proposed approach.

A. Data collection:

The first step is the collection of opinion data; we can get the data from various online sources such as Twitter, Facebook, Pinterest, Rotten Tomatoes. In this system, we required data to be labeled as positive, negative, and neutral for training and result analysis. We obtained a part of the data from Bing [5], which has a collection of product reviews (camera, phone, etc.) acquired from Amazon.com.

B. Preprocessing:

The opinion sentences may contain non-sentimental data such as URLs, hashtags "#", annotation "@", numbers, stop words. Preprocessing is the vital step to remove the noisy data.

The preprocessing module filters data and removes unnecessary characters, converts alphabets to lower case, and splits up sentences into individual words.

(1) Erratic casting: In order to tackle the problem of posts containing various casings for words (e.g., \BeAUtiFuL "), the system transforms all the input words to lower case, which provides consistency.

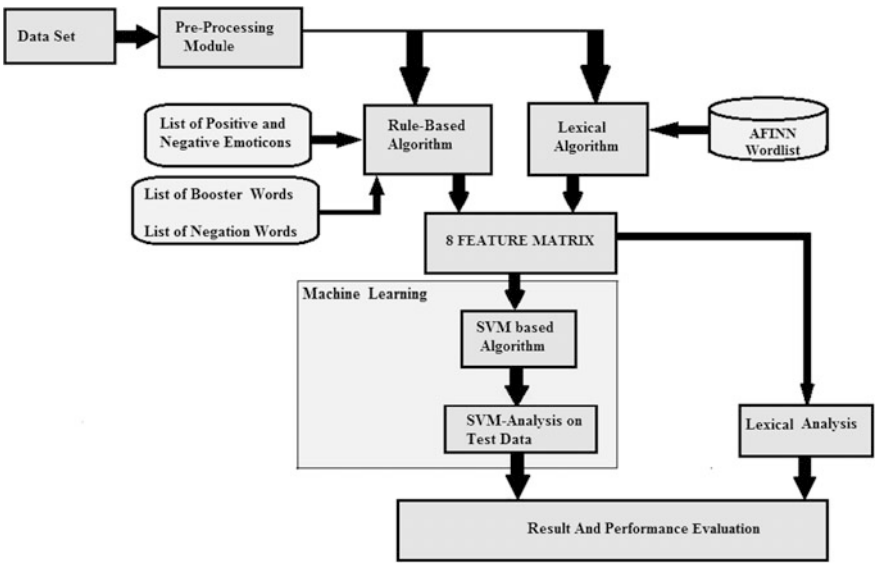

Fig. 1 Architecture of the proposed system

(2) Stop words: The presence of common words and stop words such as "a", "for", "of", "the" is usually ignored in information retrieval since their presence in a post does not provide any useful information in classifying a document.
(3) Language detection: Using NLTK's language detection feature, all sentences can be separated into English and non-English data.
(4) Split sentence: The given opinion sentence is split into separate individual words, and then, the lexical analysis is performed.

C. Rule-based Algorithm

In the rule-based algorithm, a list of positive and negative emoticons, booster words, and negation words is provided. The training data is compared with these lists and match is done. The frequencies of each of these words are obtained and used as aspect for machine learning algorithm.

(1) Emoticons: It is a symbolic illustration of a facial expression made with letters, punctuation marks, and numbers generally included in text online [21]. Many people make use of emoticons to convey emotions, which makes them very helpful for sentiment analysis [4]. A collection of about 30 emoticons, including "=[", "=)", ":)", ":D", "=]", ":]", ":(", "=(" are categorized into positive and negative.
(2) Negation words: The data contains negation words such as "no", "not", "didn't", which have a huge effect on the polarity of the sentence [9]. These words when present with a sentiment word reverse the polarity of the sentiment. For example, "I didn't like the movie" has a negative orientation even though it contains positive sentiment word "like."
(3) Booster words: Words such as "very", "highly", "most" are known as booster words; these when combined with other sentiment words enhance or boost the expression. For example, "The girl is very beautiful" gives a high positive sentiment score.

D. Lexical Algorithm

This algorithm is based on the hypothesis that the polarity of the sentence equals the summation of the polarity of individual words [7]. It uses a dictionary of positive and negative opinion words based on which the polarity of each word is obtained. In this research work, AFINN [8] wordlist is used in order to analyze the sentence content. AFINN wordlist contains about 2475 words and phrases which are rated from very positive [+5] to very negative [−5]. The words which have the score between +3 and +5 are categorized as very positive, the score of +1 to +2 categorized as positive, the score of −1 to −2 grouped as negative, and the score of −3 to −5 grouped as very negative.

The frequencies of the opinion words in each of the four categories are calculated and stored as a matrix. Also, the positive and negative emoticon frequency obtained from the rule-based algorithm is taken. Then, the basic algorithm for lexical analysis is applied, wherein the resultant polarity of the sentence is equal to the difference between the sum of positive and negative opinion words and emoticons. If the value is positive, then the sentence has positive sentiment; if the score is negative, then the polarity is negative or else the sentence is neutral.

E. Machine Learning Module

In the previous step, two outputs are obtained: the lexical analysis of the dataset as positive, negative, or neutral and a 7-feature matrix. This matrix consists of the frequencies of very positive words, very negative words, positive words, negative words, positive emoticons, negative emoticons, negation words, and booster words. It acts as a base for feature selection in machine learning. Machine learning as already discussed trains the machine on the training dataset and then tests it on the testing dataset. In machine learning, labeled examples are employed to learn classification. The machine learns by using features obtained from these examples.

Support vector machines (SVMs) are supervised techniques combined with learning algorithms that inspect data intended for categorization. Support vector machines show high efficiency at conventional data classification, usually better than Naive Bayes [4]. In the bi-class case, the basic idea behind the training procedure is to find a maximum margin hyperplane, represented by vector w, that not only separates the document vectors in one class from those in the other, but for which the separation, or margin, is as large as possible.

F. Result Analysis

The last step is the analysis of results obtained by pure lexical and hybrid approach (Fig. 2), which can be performed by evaluating confusion matrix, accuracy, recall, and performance estimation.

4 Results

The results of both of the existing classifier and proposed hybrid classifier are presented and compared in the format of deployment. Generally, performance is estimated by employing three indexes: accuracy, precision, and recall. The fundamental method for calculating these relies on confusion matrix. A confusion matrix is a matrix utilized to evaluate the performance of a classification model on a set of test data for which the true values are known [7]. It can be represented as given in Table 1.

Fig. 2 Algorithm proposed

Algorithm Proposed

1) Input Labeled Review Sentences *R*
2) Perform Pre-processing to filter noisy data, remove numbers, stop words, non-English words. Convert to lower case.
3) For each *R*, separate each word *W*, *do*
   # Lexical Analysis
   1. Input list of positive and negative sentiment words
   2. Initialize *Score, PosWord, NegWord, VPos,VNeg,* variables to *Null*;
   3. Compare each word to list of sentiment words.
   4. Get the 4-feature matrixes containing frequency of very positive, positive, very negative and negative words in the sentences.
   5. Calculate *Score* by finding difference between the frequency of positive and negative words in the sentence.
   6. Determine the polarity of the review based on the orientation of *Score*
4) *Rule-Based Algorithm*
   1 Input Rule list of Positive Emoticons, Negative emoticons, Negation Words, Booster Words
   2 Initialize Variables *PosEmo, NegEmo, BoosterW, NegW*
   3 Compare each *W* in *R* to the Rule list
   4 Obtain 4-feature matrix containing frequencies of *PosEmo, NegEmo,BoosterW, NegW*
5) Combine the Matrix from lexical and Rule based Algorithm to form 8-feature Matrix
6) For each tuple *T* in the 8-feature Matrix, do
   #Train the Machine using *Hybrid Algorithm* on Training dataset
7) Test the algorithm on Test Dataset
8) Analyze Result

Accuracy is the section of all truly predicted cases aligned with all predicted cases [22]. The accuracy of the hybrid system is found to be around 93%, which is higher than pure lexical and SVM algorithm. Precision is the segment of true positive predicted instances against all positively predicted instances [22] (Tables 2, 3, and 4).

Sensitivity/recall is the portion of true positive predicted instances next to all actual positive instances [22]. As per Tables 5 and 6, precision and recall both show significantly higher values than the pure lexical algorithm. Figure 4 is a histogram

Table 1 Confusion matrix

#	Actual positive	Actual negative
Predicted positive	Number of true positive instances (TP)	Number of false negative instances (FN)
Predicted negative instances	Number of false positive instances (FP)	Number of true negative instances (TN)

Table 2 Confusion matrix for lexical analysis

Lexical analyzer	Actual			
Predicted		Negative	Neutral	Positive
	Negative	172	16	3
	Neutral	13	173	3
	Positive	15	11	194

Table 3 Confusion matrix for hybrid system

Hybrid system	Actual			
Predicted		Negative	Neutral	Positive
	Negative	**186**	7	9
	Neutral	10	**183**	2
	Positive	4	10	**188**

The bold numbers specify the true positive cases

Table 4 Accuracy of the hybrid algorithm

	Algorithm	Accuracy
1	Lexical analysis	0.89
2	Hybrid SVM	0.93

Table 5 Precision

	Algorithm	Negative	Neutral	Positive
1	Lexical analysis	0.900	0.915	0.881
2	Hybrid SVM	0.953	0.93	0.938

Table 6 Recall

	Algorithm	Negative	Neutral	Positive
1	Lexical analysis	0.860	0.865	0.970
2	Hybrid SVM	0.930	0.915	0.980

that shows the frequency of opinion sentences having particular sentiment score. The frequency of sentences with neutral (exactly zero) orientation is maximum. The range of sentiment strength varies from −4 to +5 (Fig. 3).

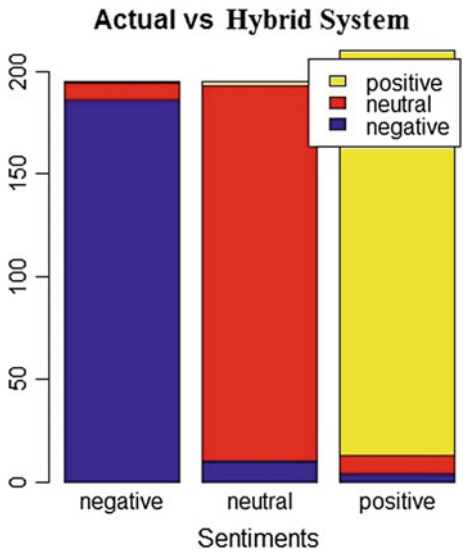

Fig. 3 Group plot for actual versus hybrid system

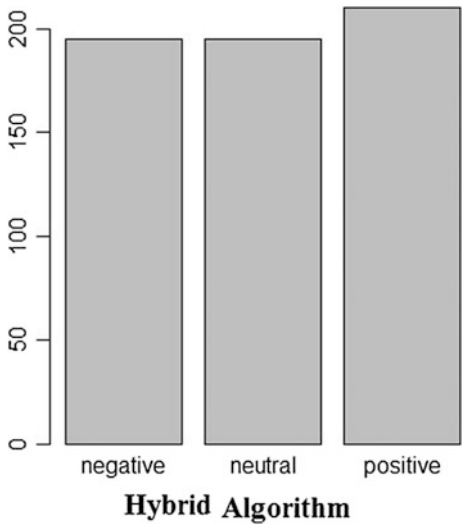

Fig. 4 Frequency of positive, negative, and neutral sentences as per hybrid technique

5 Conclusion and Future Work

For sentiment classification, the lexical-based method is a straight forward, feasible, and handy process, which does not require training dataset. Inclusion of negation words improves performance and can be further enhanced by including features such as emoticons, booster words, *n*-grams, idioms, phrases. The concept of unsupervised

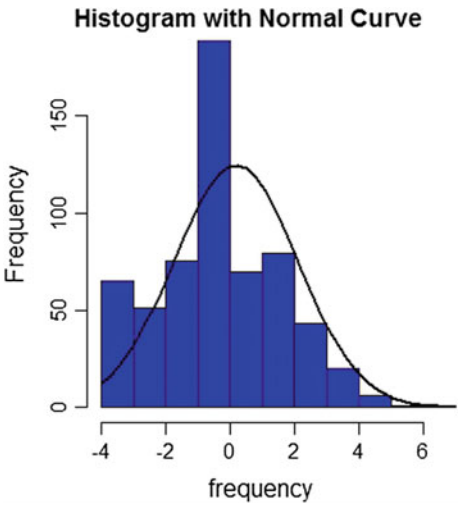

Fig. 5 Frequency of positive, negative, and neutral sentences as per hybrid technique

algorithms looks quite interesting, particularly to non-professionals. However, the performance of these algorithms is much less than those of supervised algorithms. On the other hand, supervised methods are less structured and less readable and depend greatly on the type of training data. Thus, the proposed algorithm has the ability to merge the best of the two concepts: the readability and stability. Also, this hybrid model is useful in the cross-styled environment where training is done on one dataset and testing on a different type of dataset, making the system more flexible and fault-tolerant. Hence, our approach can take over the cross-style stability of the lexical algorithm and has more accuracy of the supervised learning method. The output of the analysis can be obtained in the graphical form, which is easy to perceive and study (Fig. 5).

The system efficiently works for simple sentences; however, there are still numerous challenges that need to be researched. The hybrid system does not classify sarcasm, comparative sentences, n-grams, and multilingual sentences. It cannot detect spam opinions, grammatical mistakes, and spelling errors. The accuracy of the system can further be enhanced by adding n-gram concept and optimizing the algorithm.

References

1. B. Pang, L. Lee, Opinion mining and sentiment analysis. Found. Trends Inf. Retrieval **2**(1–2), 1–135 (2008)
2. A. Pak, P. Paroubek, Twitter as a corpus for sentiment analysis and opinion mining, in *Proceedings of LREC* (2010), pp. 1320–1326
3. D. Mumtaz, B. Ahuja, Sentiment analysis of movie review data using Senti-Lexicon Algorithm, in *Presented in ICATCCT 2016, IEEE Conference* (SJBIT Bangalore, 2016)

4. A. Mudinas, D. Zhang, M. Levene, Combining lexicon and learning based approaches for concept-level sentiment analysis, in *Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining* (ACM, New York, 2012)
5. B. Pang, L. Lee, Opinion mining and sentiment analysis. Found. Trends Inf. Retrieval. **2**(1–2), 1–94 (2008)
6. J. Rennie, L. Shih, J. Teevan, D. Karger, Tackling the poor assumptions of Naive Bayes classifiers, in *ICML* (2003)
7. D. Mumtaz, B. Ahuja, A lexical approach for opinion mining in twitter, MECS/ijeme.2016.04.03
8. F.A. Nielsen, A new ANEW: evaluation of a word list for sentiment analysis in microblogs, in *Proceedings of the ESWC2011 Workshop on 'Making Sense of Micro-posts: big things come in small packages 718 in CEUR Workshop Proceedings* (2011), pp. 93–98. http://arxiv.org/abs/1103.2903
9. R. Prabowo, M. Thelwall, Sentiment analysis: a combined approach. J. Info Metr. **3**, 143–157 (2009)
10. B. Pang, L. Lee, A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, *ACL '04* (Stroudsburg, PA, USA, 2004)
11. J. Read, Using emoticons to reduce dependency in machine learning techniques for sentiment classification, in *Proceedings of the ACL Student Research Workshop, ACL student '05* (Stroudsburg, PA, USA), pp. 43–48
12. M. Hu, B. Liu, Mining and summarizing customer reviews, in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *KDD '04* (ACM, New York, NY, USA, 2004), pp. 168–177
13. M. Hu, B. Liu, Opinion feature extraction using class sequential rules, in *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs* (Stanford, CA, USA, 2006), pp. 61–66
14. Y. He, Incorporating sentiment prior knowledge for weakly supervised sentiment analysis. ACM Trans. Asian Lang. Inf. Process. (TALIP), **11**(2), 4–1 (2012)
15. C. Lin, Y. He, Joint sentiment/topic model for sentiment analysis, in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, *CIKM '09* (ACM, New York, NY, USA, 2009), pp. 375–384
16. D. Davidov, O. Tsur, A. Rappoport, Enhanced sentiment learning using twitter hashtags and smileys, in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING'10* (Association for Computational Linguistics, Stroudsburg, PA, USA, 2010), pp. 241–249
17. A. Agarwal, B. Xie, I. Vovsha, O. Rambow, R. Passonneau, Sentiment analysis of twitter data, in *Proceedings of the Workshop on Languages in Social Media, LSM '11* (Association for Computational Linguistics, Stroudsburg, PA, USA, 2011) pp. 30–38
18. N. Mittal et al., A hybrid approach for twitter sentiment analysis, in *10th International Conference on Natural Language Processing* (*ICON-2013*) (2013)
19. N. Malandrakis et al., Sail: Sentiment analysis using semantic similarity and contrast. *SemEval 2014*, (2014), p. 512
20. Thakkar, Harsh Vrajesh, Twitter sentiment analysis using hybrid Naïve Bayes
21. https://en.wikipedia.org/wiki/Emoticon
22. jmisenet.com