Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 11, No. 2, June 2023, pp. 402~415
ISSN: 2089-3272, DOI: 10.52549/ijeei.v11i2.4187

Techniques for Improving the Performance of Unsupervised Approach to Sentiment Analysis

Farha Naznin[1], Anjana K. Mahanta[2]
[1,2]Department of Computer Science, Gauhati University, India

Article Info

Article history:

Received Sep 29, 2022
Revised Mar 24, 2023
Accepted Apr 15, 2023

Keywords:

Sentiment analysis
Clustering
Ensemble learning
Unsupervised technique
K-means algorithm

ABSTRACT

In this work, few techniques were proposed to enhance the performance of unsupervised sentiment analysis method to categorize review reports into sentiment orientations (positive and negative). In review reports, generally negations can change the polarity of other terms in a sentence. Therefore, a new technique for handling negations was proposed. As it is seen that, the positions of terms in a report are also important i.e. the same term appearing at different positions in a report may convey different amount of sentiments. Thus, a new technique was proposed to assign weights to the terms depending on their positions of occurrences within a review. Again, another technique was proposed to use the presence of exclamatory marks in the reviews as the effects of exclamatory marks are equally important in categorizing review reports. After incorporating all these concepts in the first phase of the proposed method, in the second phase, analysis of sentiment orientations was done using cluster ensemble method. The proposed method was tested on a state-of-the-art Movie review dataset and 91.75% accuracy was achieved. A significant improvement over some of the unsupervised and supervised methods in terms of accuracy was achieved with incorporation of the new techniques.

Corresponding Author:

Farha Naznin,
Department of Computer Science,
Gauhati University,
Jalukbari, Guwahati, Assam, India.
Email: farha.gu@gmail.com

1. INTRODUCTION

Sentiment analysis is a process which can be used to determine the sentiment or the opinion associated with a text. In this process, natural language processing and text mining techniques are used to find out whether the text conveys positive, negative or neutral sentiments. With the rise of blogs, micro-blogs, forum-discussions, comments, reviews posted in social media networks and websites, there is a large repository of such data. Customer's opinions help organizations while making decision about their products. That is why it is important to automate sentiment analysis. Basically, two research directions are there in sentiment analysis. One direction is categorization of the reviews into sentiment orientations such as positive, negative or neutral [1]. The second one is to extract the subjectivity or objectivity of the reviews [2].

Among the various methods used in the field of sentiment analysis, symbolic techniques and supervised techniques are those that are most commonly used [3]. Symbolic techniques need no human participation but prior studies show that the accuracy rates that are obtained using this technique are relatively low [4]. Again, in the studies done in the papers [5-11], mainly supervised techniques are used. The supervised techniques obtain relatively higher accuracy but they need human participation. Also, these methods have drawbacks like domain dependency problem and labeling cost problems. Especially, in today's era, very large and complex amounts of data from various platforms are collected. Often, these data are not labeled and a significant effort is typically required to label the data by individuals with domain knowledge.

Therefore, it becomes useful to use clustering techniques in sentiment analysis since it is efficient and being an unsupervised approach to machine learning, it does not need manual labeling of data.

In comparison to supervised methods, less works has been found in the literature in the area of sentiment analysis using clustering techniques. Here some of the notable works in this are are mentioned. Li *et al*. [4] presented an unsupervised clustering-based technique using K-means algorithm on movie review dataset. To deal with the instability of the K-means algorithm, a voting mechanism was used. To increase the accuracy rate they took adjectives and adverbs as features and used TFIDF weighting scheme. They also added the method by J. Kamps *et al*. [12] for finding the term score using WordNet [13].

AL-Sharuee *et al*. [14] introduced an unsupervised method which was based on clustering. The method proposed by us is based on the technique presented in this paper. For the sake of completeness, we describe the main features of this paper. In this paper, the work proposed can be divided into two main steps. The initial step is contextual analysis. In this step, an automatic contextual analysis is done to handle commonly occurred language related problems like negation, contrast, intensifiers etc. They handled intensifiers and negation using SentiWordNet 3.0, which is a sentiment dictionary [15]. Second step is binary ensemble clustering. In this phase, they assembled the results of clustering using different weight schemes, where the K-means algorithm was used for the clustering purpose which was modified by using SentiWordNet 3.0 to create two initial seeds for the K-means algorithm.

Another notable research was done by Riaz *et al*. [16] where, sentiment analysis was applied on a customer review dataset to detect the behavior and preferences of a customer. To deduce the intensity of the expressions, the strength of sentiment words present in an expression was computed. Then clustering was performed to position the words in different clusters according to their intensity. In another work by U. Rahardja *et al*. [17], sentiment opinions are analyzed to find out the opinions of users on an e-commerce website. Using k-medoid clustering algorithm, this method analyzed text reviews obtained from customers on the e-commerce website. In a latest study by Mohan Kumar AV *et al*. [18], ROCK algorithm is applied for clustering purpose and CART is applied to classify the positive and negative words in the comments posted by the viewers where the percentages of occurrence of both the groups of words are calculated. Finally, the method categorizes the movie which got the highest percentage of positive reviews from the users. It is observed that in most of the works based on unsupervised techniques, the accuracy rates are not as satisfactory as with the supervised methods.

The aim of the proposed method is to present a balanced unsupervised method, performance of which is competitive to that of supervised methods in terms of accuracy and being unsupervised in nature, human participation is also not needed. For this purpose, the proposed method tries to extend the work proposed in [14] for sentiment analysis by adding three new procedures in the first phase of the method that is the pre-processing phase. The first proposed technique is a novel negation handling method. Different techniques are used for negation handling in different studies such as [19-21] in order to improve the accuracy of the proposed methods. In the literature most of the papers have used traditional way for negation handling that is reverse polarity method, where the polarity of the negated word or phrase has been reversed. In some studies, such as in [22-24], some different techniques instead of just reversing the polarity have been employed. But the method proposed in this study is different from these works. In this negation handling method some special cases are considered to change the polarity of the other terms in a sentence when a negation word is encountered in the sentence, since in some special cases, negation words may not express exactly opposite of the adjectives or adverbs in the sentence. To the best of our knowledge, these types of special cases are not explored previously which may have positive implecation in the decision process.

In the second proposed procedure, if a particular sentence contains exclamation mark then weight of all the words present in the sentence has been increased as the sentences which contain exclamation marks may carry more sentiment. In studies done in [25, 26], the information carried by exclamation mark was used. But in our method the approach is different from these works.

The third proposed technique tries to improve the accuracy of the unsupervised methods by assigning weights to the words conveying sentiments using their context in the review. Usually, words appearing in some particular position in a review, may express strong feelings than those appearing in other parts of the comments [27]. For example, the first sentence in a review expresses the very first feeling of the reviewer without any self compromise being done. It has been observed that in most existing methods position context is not well explored. Usually, the fact is neglected that the position information is also crucial for identifying the sentiment polarity. In some of the notable studies that use position information of words includes the work done in [28, 29]. However, the proposed method to use position information in this work is different from these studies.

In the ensemble clustering phase, the performance of the ensemble learner proposed in [14] has been improved by applying K-Means algorithm using two different distance measures and different

representations of data. The studies done in [30-32] suggested ensemble learning method where the outputs of different vector space models and different classifiers were combined to achieve a better accuracy rate.

The main contributions of this research are-

- Designing an unsupervised clustering based algorithm to perform sentiment analysis by employing ensemble learning. In ensemble learning K-Means clustering acts as base clustering algorithm. The K-Means algorithm is applied on different representations of data using two different distance measures- Cosine distance and Jaccard coefficient. As the Jaccard coefficient is used for fuzzy sets, in this proposed method the Jaccard coefficient has been extended to numeric attributes.
- A novel method has been proposed for negation handling.
- A novel method has been proposed to use the presence of exclamatory marks in the reviews to categorize the reviews by increasing weights of the words in sentences where there is an exclamation mark (!).
- A novel method has been proposed for giving weights to the words according to their positions in the documents.

The paper is organized as follows-

The proposed method and the concepts related to the work are described in section 2. In section 3, the results that have been derived from the experiments using the proposed method are reported and a comparative analysis with some recent methods is presented. In section 4, the paper is concluded with an outline of the future scope of this research.

2. RESEARCH METHOD

The aim of the proposed method is to improve the performance of unsupervised sentiment analysis technique proposed in [14]. The proposed method presented in this work, shown in Figure 1, can be divided into two major steps. The first step consists of pre-processing of the data in order to construct the vectors for representing the documents.

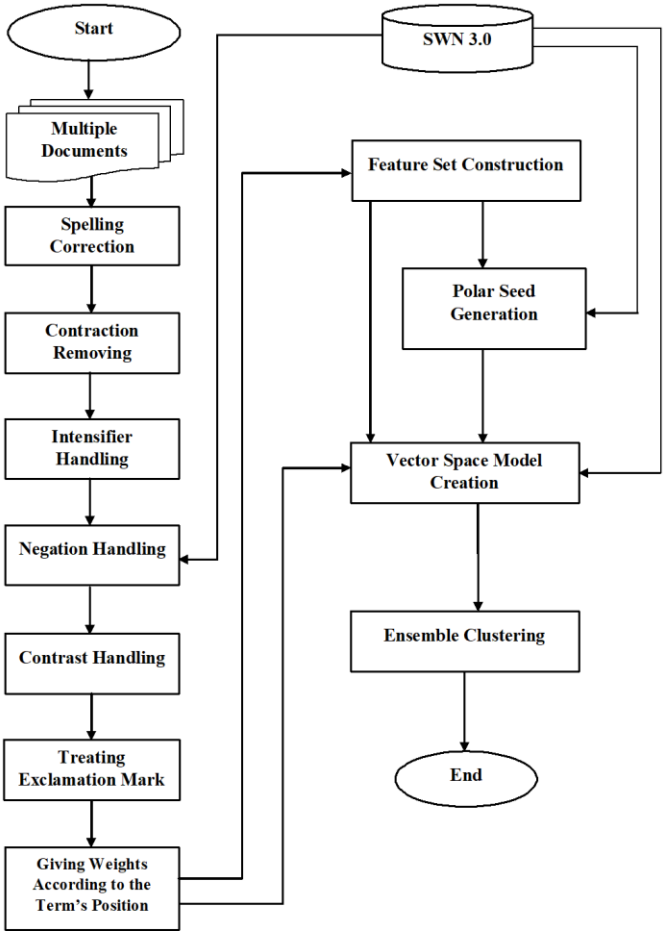

Figure 1. Flow chart of the proposed method

The next major step is ensembled based clustering in which K-Means clustering algorithm acts as base algorithm. The base algorithm K-Means operates on different data representations using different distance measures. It is already known that the initial seeds have a significant impact on the performance of K-Means algorithm. Therefore, instead of random initialization, a sentiment lexicon, SentiWordNet 3.0 [15] has been employed to construct the initial seeds for K-means algorithm. The detailed illustration of the proposed method has been given as follows-

2.1. SentiWordNet

The SentiWordNet 1.0 is a lexical resource publicly available for research purposes [33]. It is a sentiment lexicon associating sentiment information to each synset from WordNet. For each synset *s* of WordNet following scores are available in SentiWordNet-

- Positive Score *Pos(s)*
- Negative Score *Neg(s)*
- Objective Score *Obj(s)*

Each of the three scores (positive, negative, objective) is assigned a value such that for each synset *s* of WordNet the sum of these values is 1.0

$$\mathrm{Pos(s)} + \mathrm{Neg(s)} + \mathrm{Obj(s)} = 1 \quad (1)$$

Since the same word in SentiWordNet may appear in several synsets, it can have different scores in different synsets. In the proposed method SentiWordNet 3.0 has been used.

From SentiWordNet 3.0, a lexicon *U* has been constructed, where lexicon *U* contains only the adjectives and adverbs from the SentiWordNet. This concept to construct lexicon *U* is taken from paper by AL-Sharuee *et al*. [14]. In this process, for each term *I* in the SentiWordNet lexicon, it is checked if it is an adjective or adverb. If it is, then the synsets in the SentiWordNet lexicon that contain that term are identified. Then the positive scores from the synsets for that term are extracted and the average value, *avgpos(i)* of these scores is calculated. Similarly, for the negative scores also, average of negative scores, *avgneg(i)* is calculated. At the end, the score of a SentiWordNet's term *I*, *vscore* is computed by employing the following equation [14].

$$vscore\ (i) = avgpos(i) - avgneg(i) \quad (2)$$

The method that is used to build the lexicon *U* from SentiWordNet can be found in [14].

2.2. Preprocessing of the data

This is the first phase of the proposed method. In this phase the various processes are carried on the input that is, multiple text documents to prepare data for analysis in the next phase. Each document of the input contains raw plain text for performing sentiment analysis.

As the first task in this work, the spelling correction is done on the documents by using TextBlob library in python. After this, contraction removing is done on the text data. Contraction is a word or phrase which is shortened by dropping one or more letters. In this step contractions in the data files are removed by replacing them with their original words by applying a code written in python language where "re" module of python language is used. Next, intensifier handling is done on the data. In a sentence**,** Intensifier is an adverb that enhances and gives additional emotional context to an adjective. In this work, a technique is used to handle intensifiers, which is derived from [14] and implemented by using a python code. After this negation handling is done on the documents by applying a proposed technique. Then contrast handling is done on the documents. When a contrast word is used in a sentence then it means that meaning of the part which follows the contrast word in the sentence is a contradiction to the meaning of the part of the sentence that precedes the contrast word. We have derived the method for contrast handling from [14] where the part of the sentence that appears after the contrast word in the sentence is kept only and the part that appears before the contrast word is eliminated from the sentence. Then the method for treating exclamation mark is done on the dataset which is proposed in this work. Then punctuation marks are removed from the documents using "re" module of python language. After this the method for Giving Weights According to the Terms' Position is applied on the documents.

In the pre-processing phase of this work, techniques for the three methods mentioned above which are negation handling, Treating Exclamation Mark and Giving Weights According to the Terms' Position are proposed in this work. We describe these three methods below in detail-

- **Negation Handling**

Negations generally can change the polarity of other terms in a sentence. However, sometimes, it may not express the exactly opposite of the adjective or adverb. For example, "not good" may not express "bad". In this example, it may be "not good" but it may not be "bad" in total however. Moreover, there may be other cases also in case of negations, for example, "not so good", "not that good", "not good at all" etc.

In this work, a unique technique for handling negations of such cases has been proposed. The lexicon *U* generated using SentiWordNet (as given in section 2.1) is used here to handle negation. When we get a negation term in a sentence, we extract the *vscore* value of the adjective or adverb. After that, we calculate the 25% of that *vscore* value as "count" and extract the word from lexicon *U* that has a negative value of "count" if there is a "so" or "that" associated with the negation word. Then we remove the negation word along with "so" or "that" from the text and replace the adjective or adverb by the word taken out from the lexicon *U*. Similarly, if we get negation word alone, we apply the same case that with above case, just take 50% of the *vscore* value in this case. But whenever, we get "at all" associated with "not", we just replace the adjective or adverb by the word extracted from the lexicon *U* that has a negative value of the *vscore* of the adjective or adverb. This method is implemented using a piece of python code.

The method that is used for Negation handling is given below-

Algorithm 1: Negation Handling
INPUT: The dataset D containing documents obtained from the step Intensifier Handling
OUTPUT: The dataset D containing documents obtained after applying Negation Handling method

1. **for all** document $d_k \epsilon D$ **do**
2. **for all** word $w_l \epsilon d_k$ **do**
3. **if** w_l is a negation word then
4. **if** w_{l+1} is "so" or "that" **then**
5. Replace the adjectives or adverbs from w_{l+2}, w_{l+3}, w_{l+4}, w_{l+5}, w_{l+6} by the words from lexicon U that has negative value of 25% of the *vscore* values of the adjectives or adverbs.
6. else
7. **if** there is "at" and "all" with the negation word **then**
8. Replace the adjectives or adverbs from w_{l+1}, w_{l+2}, w_{l+3}, w_{l+4}, w_{l+5}, w_{l+6} by the words from lexicon U that has negative value of the *vscore* values of the adjectives or adverbs.
9. **else**
10. Replace the adjectives or adverbs from w_{l+1}, w_{l+2}, w_{l+3}, w_{l+4}, w_{l+5}, w_{l+6} by the words from lexicon U that has negative value of 50% of the *vscore* values of the adjectivesor adverbs.

- **Treating Exclamation Mark**

The *exclamation mark ("!")*, is normally used in a sentence to express strong emotion. For example, the comment with an exclamation mark "I love it!" indicates more strength than the comment "I love it.". In our work, we propose a technique where we deal with this issue by increasing the weight of the words preceded by an exclamation mark. While processing each sentence in the document, if a sentence contains an exclamation mark, then each word in that sentence are written twice, thus weight for those words are doubled. The method is implemented using a code written in python language.

The method that is used for Treating Exclamation Mark is given below-

Algorithm 2: Treating Exclamation Mark
INPUT: The dataset D containing documents obtained from the step Contrast Handling
OUTPUT: The dataset D containing documents obtained after applying the Treating Exclamation Mark method

1. **for all** document $d_i \epsilon D$ **do**
2. **for all** sentence $s_j \epsilon d_i$ **do**
3. **if** there is an exclamation mark in s_j **then**
4. Write all the words $w_k \epsilon s_j$ twice in the document d_i

- **Giving Weights According to the Terms' Position**

Words in certain position in reviews can carry more sentiment or weight than appearing elsewhere. For example, words appearing at the starting and at the end carry more sentiment than the words appearing elsewhere. Generally, in the reviews, the sentences expressing excitement appears at the beginning such as "great!" and the sentences that summarizes the whole review occurs at the end. So we are testing the effect of this case on sentiment analysis. Therefore, in this proposed method for each document the words appearing in the beginning and at the end of the document are written twice to increase their weight. Suppose, in a review

there are some words which is meaning negatively, but the summary of the review should be dependent on the few positive words present in the end of the review. In this case, the proposed method gives more weight to the positive words present at the end of the review. Thus, the proposed method will produce the analysis for the review as positive. In this method, a threshold value is used to specify what percent of total words are taken from the beginning and the end to be considered for increasing the weight.

The method that is used for Giving Weights According to the Position of the Terms is given below-

Algorithm 3: Giving Weights According to the Terms' Position
INPUT: The dataset D containing documents obtained from the step Removing Punctuation
OUTPUT: The dataset D containing documents obtained after applying the Giving Weights According to the Terms' Position

1. **for all** document $d_i \epsilon D$ **do**
2. Take a threshold value *t*
3. Calculate the value *th* as the *t* percent of the total words in d_i
4. Write twice the first *th* words and the last *th* words in the document d_i

2.3. Feature Set Construction

After applying the preprocessing steps, the feature set is constructed from the documents. In this step first adjectives and adverbs are extracted from each document as these words are sentiment indicating words [34]. After that a distinct word list of adjectives and adverbs are created from these words for all the documents. These distinct words of adjectives and adverbs are considered as features for representing the documents.

2.4. Polar seeds generation

After constructing the feature set, the polar seeds are generated which are later used as the initial points in the K-means algorithm using the features in the feature set and lexicon *U*. The method for generating the polar seeds is derived from the paper [14]. In this method, for each feature in the feature set, the value of the *vscore* of that feature is checked in the lexicon *U*. If the value of the *vscore* is greater than zero, then the feature is stored in the set of positive seed S_{pos} and if the value of the *vscore* is less than zero, then the feature is stored in the set of negative seed S_{neg}.

2.5. Vector Space Model Construction

After constructing the feature set from the documents and generating the polar seeds from the documents using the lexicon *U*, the documents and the polar seeds are converted to vectors using the vector space model so that the actual processing can be done on the documents of reviews.

The vector space model is a widely used representation for representing the text as vectors where, terms are considered as features and documents are considered as observations. After constructing the feature set, a set of matrices are created from the documents using the process derived from [14].

At first two matrices are generated using the following matrices-

a) Presence Matrix: In this representation, a binary vector is used to represent a document. In this vector, if a particular feature is present in the document, then the value will be 1 and if it is absent, the value will be 0.
b) Frequency Matrix*:* In this representation, a vector is used to represent a document. If a feature occurs c times in a document, the logarithm of c will be the value of that particular feature in the vector.

After that using these two matrices (Presence Matrix and Frequency Matrix) 10 other matrices are constructed using different weight schemes as given below-

- Term Frequency (TF)
- Inverse Document Frequency (IDF)
- Term Frequency Inverse Document Frequency (TFIDF)
- Weight Frequency-Inverse Document Frequency (WF-IDF)
- Average of weights (AW)

In addition, another 12 matrices are created, where the VSM values obtained from the former 12 matrices are increased by adding *vscore*s which are obtained from lexicon *U*.

In the construction of the vector space model, the two polar seeds (obtained by using the Polar Seeds Generation method mentioned in section 2.4) are added to the model as two documents.

The method that is used for Vector Space Model Construction can be obtained from [14].

2.6. Ensemble Clustering

The information generated by the first phase of the proposed method has been used for analysis of the sentiment orientations in this second phase. Here analysis of sentiment orientations has been done using cluster ensemble method as proposed in [14]. Ensemble learning is the process where multiple learners are joined to solve a particular problem. In the proposed method performance of the ensemble learning technique used in [14] has been improved by applying the base clustering algorithm K-means on various vector space models (as discussed in section 2.5) for two different distance measures (as discussed below). The details of this ensemble clustering have been depicted in Figure 2.

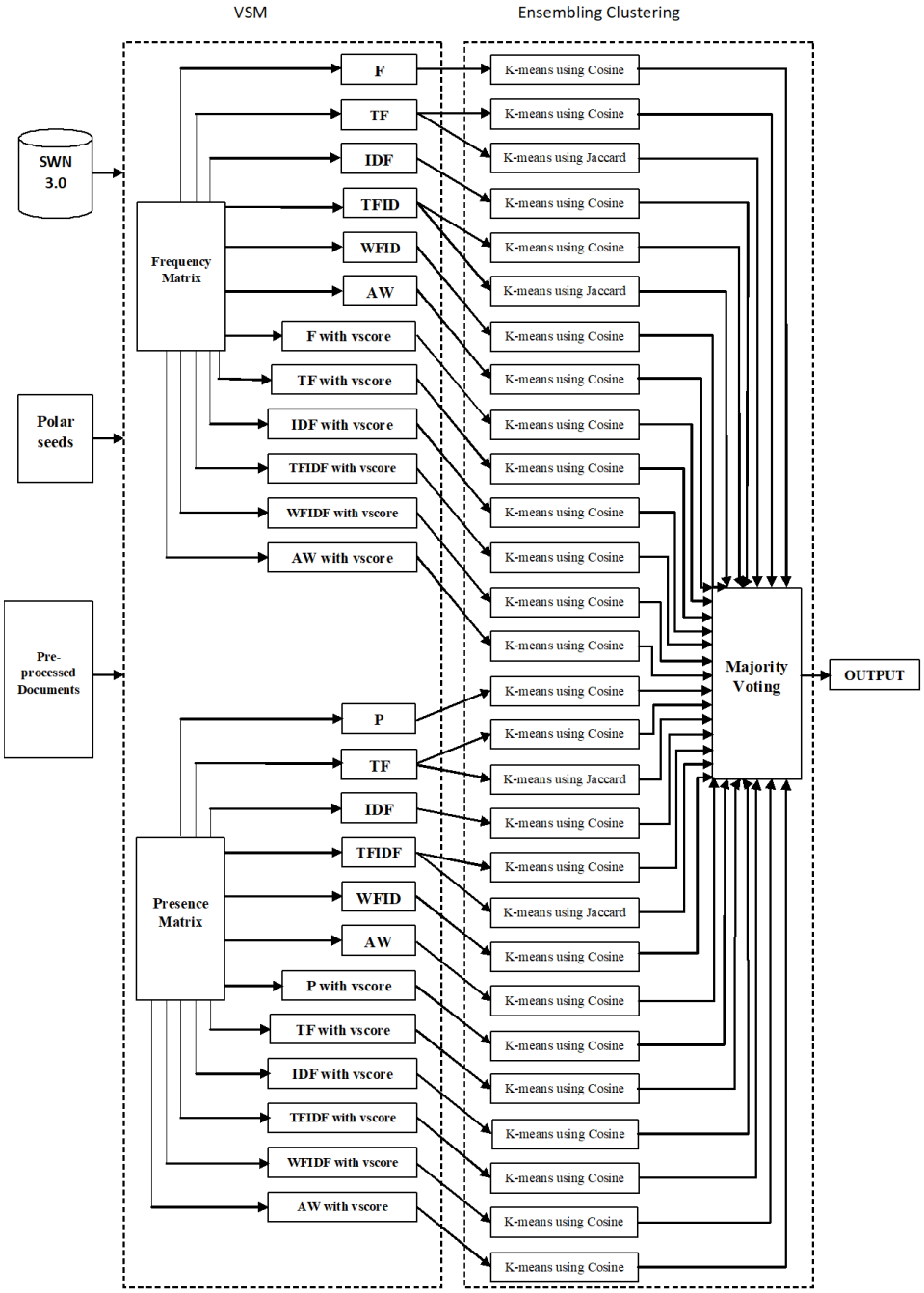

Figure 2. Framework of Ensemble clustering used in the proposed method

In the general k-means algorithm, initial centroids are selected randomly. If the randomly selected centroids are in the same cluster, then this will lead to poor clustering result. Therefore, in this work, two polar seeds *Spos* and *Sneg*, which belong to different clusters, are used as starting centroids for the base clustering method K-Means. The polar seeds are generated as given in section 2.4. These two polar seeds

Spos and *Sneg* are used to find the labels of the clusters obtained. Here, we assume that the cluster where the positive seed *Spos* occurs is the positive cluster and the cluster where the negative seed *Sneg* occurs is the negative cluster. However, in case, both the seeds occur in same group then the ensemble clustering method excludes the outcome from the final result.

This ensemble method uses two distance measures in base clustering algorithm K-Means. The distance measures are described below-

- Cosine distance**:** The Cosine distance between two vectors of attributes, *A* and *B*, is represented by:

$$\cos\theta = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \quad (3)$$

- Jaccard Coefficient for Fuzzy Sets**:** In case of crisp sets the Jaccard distance between two sets *A* and *B* is defined as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (4)$$

This definition can be extended to fuzzy sets using the corresponding definitions for union, intersection and cardinality of the sets. In TF and TFIDF representation, documents can be considered as fuzzy sets. Our proposed method applied this extension of Jaccard Coefficient for fuzzy sets on TF and TFIDF. Let *A* and *B* be two fuzzy sets representing two vectors. Then Jaccard Coefficient between the sets *A* and *B* is defined as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

Here union, intersection and cardinality of sets are their corresponding extension for fuzzy sets.

The method that is used for Ensemble Clustering is given below-

Algorithm 4: Ensemble Clustering
INPUT: A set of matrices obtained after applying Vector Space Model construction
OUTPUT: A positive or negative label for each document in the dataset D for which the vector space model was constructed.

Clustering

1. Put the number of clusters *K* equals to 2
2. **for all** matrices M_i of $M_{m(m=1,2,\ldots,8)}$, and $N_{m(m=1,2,\ldots,12)}$ **do**
3. Initialize as first centroids the positive seed *Spos* and negative seed *Sneg*
4. Apply algorithm k-means K_i with Cosine distance on M_i to produce two clusters C_1 and C_2
5. **if** *Spos* $\in C_1$ and *Sneg* $\in C_2$ **then**
6. C_1 = positive cluster, C_2 = negative cluster
7. **else if** *Spos* $\in C_2$ and *Sneg* $\in C_1$ **then**
8. C_2 = positive cluster, C_1= negative cluster
9. else
10. discard the result
11. **for all** matrix files M_i of $M_{m(m=9,10,11,12)}$, **do**
12. Initialize as first centroids the positive seed *Spos* and negative seed *Sneg*
13. Apply algorithm K-means K_i with Jaccard coefficient of fuzzy sets on Cluster M_i to produce two clusters C*1* and C*2*
14. **if** *Spos* $\in C_1$ and *Sneg* $\in C_2$ **then**
15. C_1 = positive cluster, C_2 = negative cluster
16. **else if** *Spos* $\in C_2$ and *Sneg* $\in C_1$ **then**
17. C_2 = positive cluster, C_1 = negative cluster
18. else
19. discard the result

Voting

20. **for all** document $d_l \in D$, **do**
21. **for all** result Rs_i of K_i (excluding the discarded results), **do**
22. **if** $\sum(d_l(Rs_i)=positive \geq \sum(d_l(Rs_i)=negative)$ **then**
23. d_l =*positive*
24. **else**
25. d_l =*negative*

3. RESULTS AND DISCUSSION

To test the performance of the proposed method a movie review dataset has been used. The codes are implemented with Python 3.7. In this research an HP computer with a 2.30 GHz Intel(R) Core(TM) i-5-6200U CPU, 8 GB RAM and Windows 8 operating system is used to run the programs. To evaluate the program, accuracy tests were done by building a confusion matrix using the labels positive and negative attached to each of the document. The labels that are assigned to the produced clusters can be determined by the assig$_{nm}$ent of the polar seeds S_{pos} and S_{neg}.

3.1. Experimental Data

For experimental purpose the proposed method has been applied on a movie review dataset which is previously used in many research studies in the area of sentiment analysis. This movie review dataset was first used in the paper by B. Pang *et al*. [5]. The source of the dataset is www.cs.cornell.edu. This dataset contains 2000 reviews out of which 1000 are positive and 1000 are negative. Labels are already assigned to the review documents. These are plain text data which are highly unstructured. A glimpse of the dataset is shown in Table 1. Sentiment analysis on movie review data is believed to be more difficult than any other domain [1, 35]. It is because many facets are involved here that should be considered for example, performance of the actors, script of the movie etc. We took the whole dataset in the experiment.

Table 1. A glimpse of the movie review dataset

Document	Review	Label
1	after bloody clashes and independence won , lumumba refused to pander to the belgians , who continued a...	Pos
2	the happy bastard's quick movie review damn that y2k bug . it's got a head start in this movie starring jamie...	Neg
3	it is movies like these that make a jaded movie viewer thankful for the invention of the timex indiglo watch...	Neg
4	synopsis : in this movie, steven spielberg , one of today's finest directors , attempts to spice up the 1800s...	Pos

4. RESULTS AND DISCUSSION

4.1. Results

Experiment 1: Preliminery investigation

a) Initially, in the preprocessing phase we applied the following steps– spelling correction, contraction removing, intensifier handling, contrast handling and removing punctuation mark. In the ensemble clustering phase (as described in section 2.6), K-Means clustering has been applied as base algorithm on 24 numbers of matrices using only Cosine distance. These 24 numbers of matrices are created for the data using different weight schemes like TF, IDF, TFIDF, WFIDF, AW (as described in section 2.5). With these above steps the accuracy obtained is 79.45%.

b) Now by keeping the preprocessing phase same as the above, in the ensemble clustering phase, K-Means clustering has been applied as base algorithm on 24 numbers of matrices using two distance measures-Cosine distance and Jaccard coefficient for fuzzy set (as described in section 2.6). These 24 numbers of matrices are created for the data using different weight schemes like TF, IDF, TFIDF, WFIDF, AW (as described in section 2.5). As the Jaccard coefficient is extended to be used with fuzzy sets, here it is used with only two weight schemes TF and TFIDF. When Jaccard coefficient of fuzzy sets with TF and TFIDF were used with presence and frequency matrices, we have added four more accurate and diverse members in the ensemble process, which has given a positive inference in our proposed algorithm. The accuracy obtained is 83.5%.

It is seen that the accuracy increases by 4.05% than the former experiment.

Experiment 2: Overview of the accuracy enhancement of the proposed method by adding the proposed negation handling technique

In the preprocessing phase, negation handling method is added with the other preprocessing steps mentioned in the Experiment 1. After adding this, the accuracy enhances to 82.1% with only Cosine distance and to 85.5% with the ensemble of Cosine distance and Jaccard coefficient. So with negation handling, accuracy increases 2.65% with cosine distance and 2% with the ensemble of Cosine distance and Jaccard coefficient.

Experiment 3: Overview of the accuracy enhancement of the proposed method by adding the proposed method-Treating the exclamation mark

In this experiment, in the preprocessing phase, the method treating the exclamation mark has been added in preprocessing phase along with the other preprocessing steps mentioned in Experiment 1 and the proposed negation handling method. The accuracy increases to 85.05% with Cosine distance only and 89.5% with the ensemble process of the two distance measures. Hence with this addition, accuracy increases 2.95% with cosine distance and 4% with the ensemble of Cosine distance and Jaccard coefficient.

Experiment 4: Overview of the accuracy enhancement of the proposed method by adding the proposed method- where the words are given weights according to their position is added in preprocessing

In this experiment, in the preprocessing phase, the method giving weights according to the terms' position is added in preprocessing along with the preprocessing steps mentioned in Experiment 3. When ten percent of total words are taken as threshold the accuracy further increases to 85.5%. But when we applied it with ensemble clustering with Cosine distance and Jaccard coefficient of fuzzy sets we got the accuracy as 91.5% which is a noticeable improvement over the former experiment. Adding this proposed method with 10% threshold, accuracy increases only 0.45% with Cosine distance but 2% with the ensemble of Cosine distance and Jaccard coefficient. Again when we applied twenty percent of total words in a document as threshold, first with only Cosine distance and then with ensemble clustering of Cosine distance and Jaccard coefficient, we got 85.6% accuracy with Cosine distance. With ensemble clustering of Cosine distance and Jaccard coefficient, we got 91.75% accuracy. Adding this proposed method with 20% threshold, accuracy increases only 0.55% with Cosine distance but 2.25% with the ensemble of Cosine distance and Jaccard coefficient. We got slight improvement when we use the twenty percent words of all the words in the document as threshold. With the testing of ten percent and twenty percent we got almost same results with our proposed method.

In Table 2 the accuracy of the proposed method by applying the negation handling technique, treating exclamation mark and giving weight according to the terms' position with only Cosine distance and with the ensemble of Cosine distance and Jaccard coefficient of fuzzy sets with two threshold value 10% and 20% is shown. Again, in Figure 3, the increase of accuracy of the proposed method by applying these techniques is shown. The accuracy obtained by each component of our proposed method (using all the proposed techniques) is shown in Figure 4.

Table 2. The accuracy of the proposed method obtained with ensemble clustering using only Cosine distance and ensemble clustering using Cosine distance and Jaccard coefficient of fuzzy sets using the proposed techniques added in the preprocessing phase

Proposed technique added in the preprocessing phase of the proposed method	Accuracy with Ensemble clustering using only Cosine distance	Accuracy with Ensemble clustering using both Cosine distance and Jaccard coefficient of fuzzy sets
Negation Handling	82%	85.5%
Negation Handling + Treating Exclamation Mark	85.05%	89.5%
Negation Handling + Treating Exclamation Mark + Giving Weight According to the Term's Position (Threshold=10%)	85.5%	91.5%
Negation Handling + Treating Exclamation Mark + Giving Weight According to the Term's Position (Threshold=20%)	85.6%	91.75%

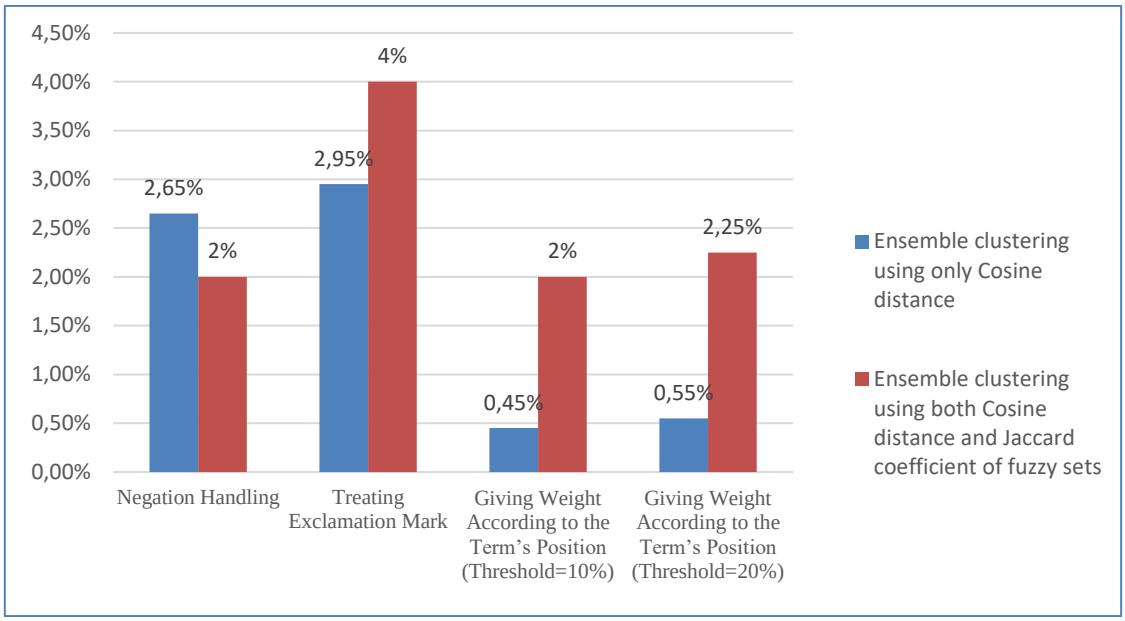

Figure 3. The increase of accuracy of the proposed method with ensemble clustering using only Cosine distance and ensemble clustering using Cosine distance and Jaccard coefficient of fuzzy sets using the proposed techniques added in the preprocessing phase

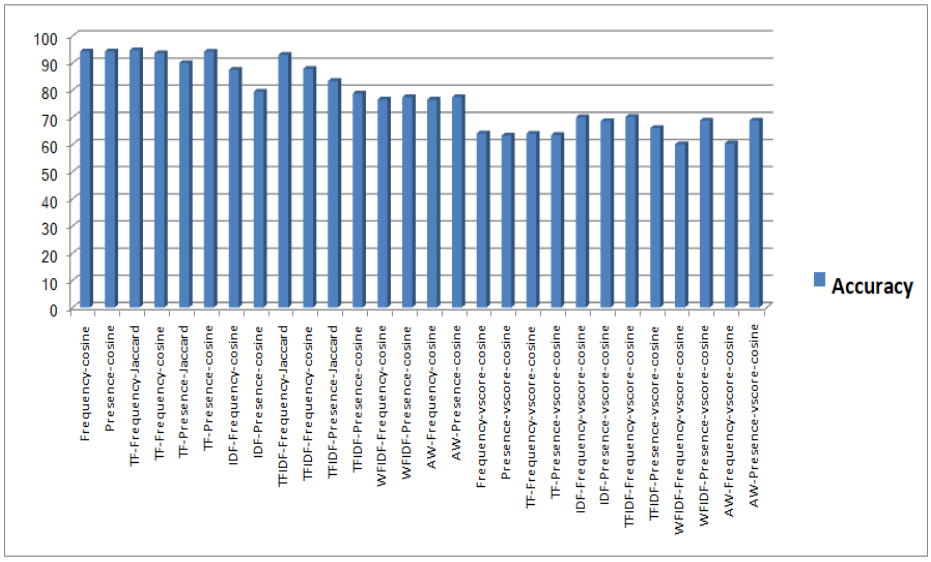

Figure 4. Performances of the ensemble components of the proposed method

Experiment 5: Experiments with supervised methods

In this experiment separate programs are run using the classification techniques- Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF), Multinominal Naïve Bayes (MNB) and Decision Tree (DT). In these programs, in the first phase that is in the preprocessing step, spelling correction, contraction removing, intensifier handling, contrast handling and removing punctuation mark are applied on the data but the three techniques proposed in the preprocessing phase in this work are not applied on the data. TFIDF weight scheme is used to represent the data. Then, the classification techniques are applied on the dataset using the scikit-learn module (version 0.21.3) of python language. The accuracies obtained with SVM, LR, RF, MNB and DT are 85%, 83%, 81.4%, 77.4% and 77% respectively. A comparison between the accuracy of these supervised techniques and the accuracy obtained with the proposed method (applying all the proposed techniques) is shown in Table 3.

Table 3. Comaprison between the accuracy obtained with supervised techniques in Sentiment Analysis and the proposed method

Method	Accuracy
SVM	85%
LR	83%
RF	81.4%
MNB	77.4%
DT	77%
Proposed method	91.75%

3.2.2. Discussion and evaluation of results:

In the Experiment 1, result shows that using Jaccard coefficient of fuzzy sets along with the Cosine distance in the ensemble process gives better accuracy than that of using Cosine distance alone in the ensemble process. So, incorporation of Jaccard coefficient of fuzzy sets along with the Cosine distance in the ensemble process gives a significant improvement in our proposed method.

Again by adding the newly proposed preprocessing steps- negation handling, treating the exclamation mark and giving weights according to the terms' position (as described in section 2.2) in the first phase, the result of the proposed method improves (Table 2) than the method where these preprocessing steps are not added. So it provides a clear implecation that incorporating these methods in our algorithm yeilds a significant enhancement in the accuracy of the proposed method.

Moreover, from the Table 3, we can see that the accuracy of the proposed method in this work gives better accuracy than the supervised techniques for sentiment analysis on the Movie Review dataset.

In Table 4, we compare the performance of the proposed method on Movie Review dataset with the results of the existing methods on Movie review dataset proposed in other published papers. As shown in the Table 4, our proposed method gives better result than most of the methods on the Movie review dataset.

Table 4. Comparison of the accuracy of the existing methods in Sentiment Analysis with the proposed method

Author	Year	Techniques used	Accuracy
Bo Pang et al. [5]	2002	Naive Bayes, maximum entropy classification, and support vector machines	72%-82%
Gang Li et. al. [4]	2012	K-means clustering algorithm, voting mechanism, importing term scores.	77.17-78.33
M. Govindarajan [6]	2013	Naïve Bayes, Genetic algorithm	93.80%
Raj K. Palkar et. al. [7]	2016	Naïve Bayes, Support Vector Machine, Maximum Entropy, Random Forest	69%-83%
M. T. AL-Sharuee et. al. [14]	2018	Ensemble clustering using K-means algorithm with different weight schemes	80.41%
Mohan Kumar AV et. al. [18]	2019	ROCK, CART	89.76%
Atif Khan et. al. [8]	2020	bag-of-words feature extraction technique (unigrams, bigrams, and trigrams), Naïve Bayes algorithm	88.90%
Reza Maulana et. al. [9]	2020	Support Vector Machine, Information Gain	85.65%
		Our proposed method	91.75%

From the experiments it is seen that a significant enhancement in the performance of the proposed method is obtained after using the modifications such as ensemble clustering with Cosine distance and Jaccard coefficient of fuzzy sets, proposed negation handling technique, treating exclamation mark and giving weights according to the terms' position. The ensemble clustering with Cosine distance and Jaccard coefficient of fuzzy sets has positive implecation for our proposed method. Also, the proposed preprocessing techniques have significant impact on the accuracy of the proposed method. It is because one of the proposed techniques in this work which is negation handling is a common language form that should be handled. Moreover, other two techniques- treating exclamation mark and giving weight according to the terms' position also affect in the sentiment orientation of reviews or comments. We tested all these three conditions in our proposed method and these give positive implecation. From this research work it can be observed that the sentiment analysis task can be effectively handled by our proposed method and the performance of our proposed algorithm was competitive to the performances of the supervised techniques in terms of accuracy. Again, since the polar seeds that are generated by using a method derived from [14] are used as the initial

centroids in our algorithm, hence the instability problem of K-means algorithm is addressed. So our algorithm is stable.

Moreover, our proposed method is an unsupervised technique. It does not need manual labeling; hence no human participation is needed in this method. So our proposed method in sentiment analysis is a balanced method that is an unsupervised method in which we do not need human intervention and also we do not compromise with thre accuracy.

5. CONCLUSION

In this work, an unsupervised and automatic approach to sentiment analysis has been proposed. The approach used in this work is based on the work done by AL-Sharuee *et al*. [14]. In the preprocessing phase of the method a new idea for handling negations has been done. By applying this technique remarkable improvement is achieved. Also in the preprocessing phase, the effect of exclamation mark and term position has been considered. It further enhances the accuracy. In the ensemble clustering phase K-Means algorithm is used as base clustering algorithm where the cluster ensemble method has been executed for two different distance measures- Cosine distance and Jaccard coefficient on different weight schemes. After adding all the proposed techniques, with the proposed method we achieve an accuracy of 91.75% which is a remarkable enhancement over the results presented by AL-Sharuee *et al*. [14]. Also, the proposed method is stable and domain independent. So, the proposed method is a balanced method which requires no human participation and competitive to supervised techniques in terms of accuracy.

As future work the three classes problem i. e. adding a neutral class also could be taken. Aspect based sentiment analysis can also be incorporated with the concepts that have been implemented here which may lead to further improvements.

ACKNOWLEDGMENTS

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

[1] P. Chaovalit, *et al*., "Movie review mining: a comparison between supervised and unsupervised classification approach," in *Proceedings of the 38th Hawaii international conference on system sciences*, IEEE Computer Society, 2005.
[2] J. M. Wiebe, "Learning subjective adjectives from corpora," in *Conference on artificial intelligence,* Menlo Park, CA. AAAI Press, pp. 735–741, 2000.
[3] E. Boiy, *et al.*, "Automatic sentiment analysis in on-line text," in *International conference on electronic publishing pages*, Vienna, Austria, pp. 349–360, 2007.
[4] G. Li and F. Liu, "Application of a clustering method on sentiment analysis," *Journal of Information Science,* pp. 127-139, 2012.
[5] B. Pang, *et al.*, "Thumbs Up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, *Vols 10*, pp. 79–86, 2002.
[6] M. Govindarajan, "Sentiment analysis of movie reviews using hybrid method of Naïve Bayes and genetic algorithm," *International Journal of Advanced Computer Research*, vol. 3, 2013.
[7] R. K. Palkar, *et al*., "Comparative Evaluation of Supervised Learning Algorithms for Sentiment Analysis of Movie Reviews," *International Journal of Computer Applications*, vol. 142, pp. 20-26, 2016.
[8] A. Khan, *et al*., "Summarizing Online Movie Reviews: A Machine Learning Approach to Big Data Analytics," *Scientific Programming*, vol. 2020, 2020.
[9] R. Maulana, *et al*., "Improved Accuracy of Sentiment Analysis Movie Review using Support Vector Machine Based Information Gain," *Journal of Physics: Conference Series*, vol. 1641, 2020.
[10] B. Zhang, H. Zhang, J. Shang, and J. Cai, "An Augmented Neural Network for Sentiment Analysis Using Grammar," *Front Neurorobot*, vol. 16, 2022.
[11] N. J. Prottasha, *et al.*, "Transfer Learning for Sentiment Analysis Using BERT Based Supervised Fine-Tuning," *Sensors*, vol. 22, 2022.
[12] J. Kamps, *et al*., "Using WordNet to measure semantic orientations of adjectives," in *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC), Vols 4*, Citeseer, pp.1115–1118, 2004.
[13] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, "Introduction to wordnet: an on-line lexical database," *International Journal of Lexicography*, vol. 3, 1991.
[14] M. T. AL-Sharuee, F. Liu, and M. Pratama, "Sentiment analysis: An automatic contextual analysis and ensemble clustering approach and comparison," *Data & Knowledge Engineering*, pp. 194-213, 2018.
[15] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining," *Language Resources and Evaluation Conference (LREC)*, vol.10, pp. 2200–2204, 2010.

[16] S. Riaz, M. Fatima, M. Kamran, and M. Nisar, "Opinion mining on large scale data using sentiment analysis and k-means clustering," *Cluster Computing 22*, 2019.
[17] U. Rahardja, *et al.*, "Opinion Mining on E-Commerce Data Using Sentiment Analysis and K-Medoid Clustering," in *2019 Twelfth International Conference on Ubi-Media Computing (Ubi-Media)*, pp. 168-170, 2019.
[18] M. K. AV, *et al*, "Sentiment Analysis Using Robust Hierarchical Clustering Algorithm for Opinion Mining On Movie Reviews-Based Applications," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, pp. 452-457, 2019.
[19] P. K. Singh and S. Paul, "Deep Learning Approach for Negation Handling in Sentiment Analysis," *IEEE Access*, vol. 9, pp. 102579-102592, 2021.
[20] L. H. Kamal, G. T. McKee, and N. A. Othman, "Naïve Bayes with Negation Handling for Sentiment Analysis of Twitter Data," in *2022 9th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, Toronto, ON, Canada, pp. 207-212, 2022.
[21] U. Farooq, H. Mansoor, A. Nongaillard, Y. Ouzrout, and M. A. Qadir, "Negation Handling in Sentiment Analysis at Sentence Level," *Journal of Computers*, vol. 12, no. 5, 2017.
[22] S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, vol. 50, pp. 723–762, 2014.
[23] A. Muhammad, N. Wiratunga and R. Lothian, "Contextual sentiment analysis for social media genres," *K* A. Muhammad, N. Wiratunga and R. Lothian, "Contextual sentiment analysis for social media genres," *Knowledge-Based Systems*, vol. 108, pp. 92–101, 2016.
[24] I. Gupta, *et al.*, "Feature-Based Twitter Sentiment Analysis with Improved Negation Handling," *IEEE Transactions on Computational Social Systems*, vol. 8, pp. 917-927, 2021.
[25] P. L. Teh, *et al.*, "Sentiment analysis tools should take account of the number of exclamation marks!!!," in *Proceedings of the 17th International Conference on Information Integration and Web-Based Applications & Services*, 2015.
[26] Z. Q. Shen, T. Song, Q.R. Mao, and Z. Jiang, "An Emotion Feature Highlighting Method for Sentiment Analysis of Social Media Text," *Journal of Computers*, vol. 30, pp. 117-129, 2019.
[27] S. Mukherjee and P. Bhattacharyya, "Sentiment Analysis: A Literature Survey," *ArXiv*, 2013.
[28] G. Paltoglou, *et al.*, "More than Bag-of-Words: Sentence-based Document Representation for Sentiment Analysis," in *Proceedings of Recent Advances in Natural Language Processing*, Hissar, Bulgaria, pp. 546–552, 2013.
[29] X. Wang, X. Chen, M. Tang, T. Yang, and Z. Wang, "Aspect-Level Sentiment Analysis Based on Position Features Using Multilevel Interactive Bidirectional GRU and Attention Mechanism," *Discrete Dynamics in Nature and Society*, Hindawi, 2020.
[30] R. Xia, *et al.*, "Polarity shift detection, elimination and ensemble: A three-stage model for document-level sentiment analysis," *Information Processing and Management*, vol. 52, pp. 36–45, 2016.
[31] G. Wang, *et al.*, "POS-RS: a Random Subspace method for sentiment classification based on part-of-speech analysis," *Information Processing and Management*, vol. 51, pp. 458–479, 2015.
[32] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Information Sciences*, vol. 181, pp. 1138–1152, 2011.
[33] A. Esuli, *et al.*, "SentiWordNet: a publicly available lexical resource for opinion mining," in *Proceedings of Language Resources and Evaluation (LREC), Vols 6*, Citeseer, pp.417–422, 2006.
[34] F. Benamara, *et al.*, "Sentiment analysis: adjectives and adverbs are better than adjectives alone," in *International conference web-logs and social media (ICwsm 07)*, 2007.
[35] P. D. Turney, Thumbs up or thumbs down? "Semantic orientation applied to unsupervised classification of reviews," in *40th annual meeting of the association for computational linguistics (ACL)*, Philadelphia, Pennsylvania, USA, pp. 417-424, 2002.

BIOGRAPHY OF AUTHORS

Farha Naznin is a research scholar in the department of Computer Science in Gauhati University, Assam, India. She obtained B. Sc. degree in Computer Science from Gauhati University in 2009 and M. Sc. degree in Computer Science from Gauhati University in 2011. Her research interests include Data Mining, Natural Language Processing and Pattern recognition. She can be contacted at email: farha.gu@gmail.com.

Anjana K. Mahanta is working as a professor in the department of Computer Science in Gauhati University, Assam, India. She obtained B. Sc. degree in Mathematics from Gauhati University in 1981 and M. Sc. degree in Mathematics from Gauhati University in 1983. She obtained PGDCSA degree from Gauhati University in 1986. She obtained Ph. D. degree in Computer Science from Gauhati University in 1990. Her research interests include Data Mining, Design of algorithms and Pattern recognition. She can be contacted at email: anjana@gauhati.ac.in.