Ensemble of Feature Ranking Methods Using Hesitant Fuzzy Sets for Sentiment Classification

Gunjan Ansari, Tanvir Ahmad, and Mohammad Najmud Doja

***Abstract*—The increase in the volume of opinion posted on social media sites has led to a tremendous increase in the dimensionality of data used for the sentiment analysis. The selection of informative features from textual data can improve the performance of supervised learning methods. In this article, we propose a novel and efficient method for integrating different filter-based feature selection methods for sentiment classification. The ensemble method utilizes hesitant fuzzy sets for representing opinions of different filter-based feature selection methods in order to optimize the relevancy score among features and class labels. Based on this relevancy score, top-k ranked features are selected for sentiment classification. The proposed feature selection method with Naïve Bayes and Support Vector Machine classifiers was evaluated on three most widely used datasets for sentiment analysis using Unigram and Parts-of-Speech based text representation schemes. The performance is evaluated using five-fold cross validation technique and the results show that the proposed method can achieve greater value of accuracy with only 10-25% of total extracted features. The outcomes of comparison carried out via statistical tests confirm that the aggregation using hesitant fuzzy sets is more effective than baseline feature selection methods on Parts-of-Speech features in terms of performance metrics.**

***Index Terms*—Filter-based feature selection, hesitant fuzzy sets, Naïve Bayes classifier, sentiment classification, support vector machine.**

I. Introduction

The views and opinions posted by people on the World Wide Web is growing exponentially. The posted content need to be analyzed in order to help the organizations, companies or individuals for better decision making. Sentiment analysis is a research area of text mining which involves computational identification of people's opinions, attitudes and emotions towards an entity [1]. Sentiment classification is a major task of SA process which assigns positive or negative polarity to the given opinion document. It can be viewed as a problem of supervised learning from machine learning perspective. The authors in their work [2] showed that supervised learning techniques such as NB, SVM and ME outperform human-produced baseline approaches. Sentiment classification involves feature extraction and selection from textual opinions before classification of opinion given in the samples into positive or negative class.

Feature selection methods are aimed at improving the classifier performance by reducing the dimensionality of the extracted feature set in order to overcome overfitting problem.

The time complexity of the model can also be reduced by selection of more informative feature subset. The feature selection techniques can be classified into three types [3]: Filter methods, wrapper methods and embedded methods. Filter approach is also categorized as univariate approach as it selects informative features on the basis of relationship between individual feature and class. These methods use statistical characteristics of the features in the selection process. These methods are fast and useful in high dimensional datasets. The filter based feature selection discussed in their work were correlation criteria and mutual information. These methods have low computational cost and thus are suitable for high-dimensional datasets. The drawback of filter methods is that they do not involve feature interaction which may lead to selection of redundant features. Wrapper approaches are multivariate approach and they find optimal feature set by evaluating goodness of the subset based on its classifier performance. Though these approaches are computationally expensive, they select more relevant features compared to filter methods.

To overcome the problem of dependency of classifiers on individual feature selection method, we proposed a univariate method to integrate filter-based feature selection methods using hesitant fuzzy sets (HFS).The HFS has been used in our work to aggregate the opinion of different feature rankers on high dimensional feature set through a function for final decision making. The aggregated values are then used to rank the features for final feature selection.

The main contributions of the paper are as follows:

- A univariate approach to ensemble filter-based feature selection methods using hesitant fuzzy set (HFS-FS) is proposed for sentiment classification
- Performance analysis of proposed approach using NB and SVM classifiers on both POS-based and unigram-based feature representation schemes
- The experimental results are obtained on three widely used datasets of sentiment analysis domain
- A paired t-test is conducted to find statistical significance of the HFS-FS as compared to baseline algorithm for feature selection

The paper is organized as follows: Section II briefly presents related work in the field of sentiment analysis and feature selection methods. Section III gives a brief introduction on HFS, Filter based feature selection methods and classifiers used in our work. Section IV describes the methodology used to integrate feature selection methods using hesitant fuzzy set approach. Experimental results are presented in Section V. Finally, Section VI concludes the paper and scope for further enhancement in the work.

Manuscript received February 5, 2019; revised May 19, 2019.

Gunjan Ansari, Tanvir Ahmad and Mohammad Najmud Doja are with the Department of Computer Engineering, Faculty of Engineering & Technology, Jamia Millia Islamia, Jamia Nagar, New Delhi-110025, India (e-mail: gunjanansari@jssaten.ac.in, tahmad2@jmi.ac.in, mdoja@jmi.ac.in).

doi: 10.18178/ijmlc.2019.9.5.846

II. Related Work

Sentiment classification using machine learning approach was popularized in early 2000s [2]. Three popular machine learning algorithms: Naïve Bayes, Support vector machine and Maximum entropy were employed by the authors for sentiment analysis of movie data. In the unsupervised learning approach proposed [4], three steps were performed to classify review as positive or negative. In the first step, phrases containing adjective or adverb were extracted from the review text. In the next step, semantic orientation of these phrases was computed by using PMI-IR and in the last phase, the overall sentiment of the review was computed by taking the average of these phrases. The sentiment detection of reviews [5] faced two major problems: subjectivity and sentiment classification. The authors discussed about the machine learning approaches and the major issues encountered while applying them in word and document-level sentiment classification problems. In the sentiment analysis survey by [1], feature extraction/selection and sentiment classification are the two main steps in the process of sentiment analysis. The selection of appropriate features before classification can improve the performance of classifier.

Feature engineering has become one of the most important task of any machine learning algorithm. The authors [6], [7] used unigrams, bigrams, trigrams and combination of these as features with Naïve Bayes, Support vector machine, Maximum Entropy and Stochastic Gradient Descent as classifiers for the task of polarity detection of IMDB movie reviews. The work [8] utilized information gain as heuristic of genetic algorithm for selecting optimal feature set. They evaluated their approach on the benchmark dataset of movie reviews and were able to achieve good accuracy with SVM classifier. A new feature selector that used SentiWordNet with Proportionate Difference (SWNPD) and SentiWordNet with Subjectivity Scores (SWNSS) was introduced [9]. The authors also proposed new feature weighing method based on SentiWordNet for word scoring groups and polarity groups. The significant difference of their work from conventional feature selection and weighing methods was proved using NB and SVM classifier.

Rogati M. *et al.* [10] presented a comparison on different filter-based feature selection methods such as Document frequency, information gain, mutual information, Chi square and term frequency for text classification problem. They achieved best results using Chi square test and Information gain on four classifiers. Chen J. *et al.* [11] presented two feature evaluation metrics: Multi-class odds ratio and Class Discriminating measure for the Naïve Bayesian classifier that was applied on multiclass text datasets. A new feature selection method based on query expansion ranking approach used in information was proposed [12]. The authors compared their approach with baseline feature selection methods on Turkish and English product and movie reviews using Naïve Bayes, SVM, Maximum Entropy and Decision Tree classifiers. In the study [13], hybrid of feature extraction and selection for dimensionality reduction was proposed to utilize strength of both. Hybrid approaches take benefit of both filter and wrapper method to choose features with minimum redundancies and thus increase classifier performance [14].

It was found [15] that more robust feature subsets can be generated by using ensemble of feature selection methods rather than a single feature selection method. Also the ensemble methods can solve the problem of classifier dependency on only one feature selection method. Ensemble of different feature sets and classifier using fixed, weighted and meta-classifier combination as integration strategies was utilized [16] to improve the classification performance. The effectiveness of the work was tested by SVM, ME and NB classifiers on part-of-speech base and word-relation base features. The robust and efficient feature set can be selected by aggregation of individual feature sets generated by various feature selection methods. Genetic approach was utilized [17] to aggregate filter-based feature selection methods for optimal feature selection. In the work, the optimized feature list is the list that minimizes the Spearman foot rule distance between the existing feature lists. NB and K-nearest neighbor algorithm were used as base learners in the experimental study. A hybrid of filter and wrapper approach is proposed [18] in order to obtain an optimal feature set. They obtained final feature set using two approaches: Ordinal based integration of feature vector (OIFV) and frequency based integration of feature vector (FIFV). In OIFV, feature subsets were obtained using different filter based feature selection methods in the first step. In the second step, an ordinal based integration of these subsets is done to generate new feature subsets which are evaluated by four different classifiers to obtain optimal feature subset with best accuracy. In their second approach (FIFS), they used wrapper method to evaluate feature subsets obtained by different feature rankers to generate feature vectors and then applied frequency based integration on feature vector to get final feature set.

III. Preliminaries

The proposed ensemble of feature selection uses hesitant fuzzy sets, filter algorithms and classifier. The concepts of these will be reviewed in the subsections.

A. Hesitant Fuzzy Sets (HFS)

The concept of hesitant fuzzy sets and its definition was introduced [19], [20]. The hesitant fuzzy sets is defined as follows:

Definition 1: Let X be a reference set, then hesitant fuzzy set on X in terms of a function h is that when applied to X returns a subset of [0,1].

Definition 2: Let $M = \{\mu 1, \mu 2, \mu 3 \ldots\ldots\ldots \mu N\}$ be a set of N membership functions. Then, the hesitant fuzzy set associated with M is defined as follows:

$$h_M(x) = \cup_{\mu \epsilon M} \{\mu(x)\} \quad (1)$$

HFS were introduced as simplification of fuzzy sets. The modelling of decision using HFS was further discussed in their work [21]. When decision is represented by different fuzzy sets, they can be used to aggregate decision information through some function for final decision making. The aggregated values are used to rank the alternative or select few of them.

B. Feature Representation Methods

For sentiment analysis problems, feature engineering consist of two phases: first phase is feature identification and next is feature selection. The main challenge is extraction of appropriate features and then selecting most valuable features before representing a piece of text into a feature vector. The main feature representation methods as discussed [22] are as follows:

N-gram features: These features are sequence of n objects from a given sample of text at token/word or phoneme level. These features can be classified as unigram or N-grams (a sequence of two or more words).

Parts-of-speech (POS): POS information is commonly exploited in the area of sentiment analysis. POS is linguistic category of words such as Noun, Pronoun, adjective, adverb etc. The POS tagging of documents is utilized to extract adjectives in a piece of text as this feature is highly correlated with sentence subjectivity.

Negation: These features play a major role in opinion mining as they can change the semantic meaning of the given token. The bag-of-word representation of "I like this movie" and "I don't like this movie" is same but the negation used in latter changes the sentiment of the sentence.

Syntactic and semantic dependency: Documents can be parsed to extract dependency between words in sentiment classification for extracting relevant feature set. Parsing or dependency tree can be utilized for modelling valence shifters [23] such as negation, diminishers and intensifiers in text classification.

In our experimental work, we used two types of feature representation methods: fixed and variable n-gram features [18]. In the fixed n-gram, unigrams, bi-grams and tri-grams of fixed size are extracted. We used only unigram features in our work. Variable n-grams are extracted from raw document using POS tags. First POS tags are assigned to the whole sentence. Then based on the linguistic filter, POS patterns of variable length (1-3) are extracted. For detecting features as Noun, Adjective, Verb and Adverb, twenty-three POS filters are utilized.

C. Feature Selection Methods

The five global filter-based feature selection methods utilized in the proposed approach are discussed as follows:

Chi-square Test [24]: It is the global feature selection method. It is used to calculate the degree of relationship between feature and class. A higher value of feature-class score implies that class is more dependent on that feature and thus the feature is more informative. The formula used to calculate the Chi-square score of a feature is as follows:

$$CHI(f, C_k) = \frac{N(AB-CD)^2}{(A+C)(A+D)(B+C)(B+D)} \quad (2)$$

where f is feature, C_k is k^{th} class and k=1, 2…m, m is the number of classes, n is the number of documents in the dataset, A is the number of documents in which f and C_k co-occur, B is the number of documents that neither contain f nor C_k, C is the number of documents in which f occurs without C_k and D is the number of documents in which C_k occur without f.

Document Frequency [24]: Document Frequency (DF) is one of the simplest method for feature ranking in text classification. It is based on the concept that informative features occur more number of times in the documents of the corpus. DF is computed by counting the number of documents in the corpus that contain that feature f.

Information Gain [24]: Information gain (IG) is commonly utilized feature selection method for text categorization problems. It is two-sided global feature selection metric. The score of IG is obtained by the presence or absence of a term in a document for predicting the correct class of the document. The formula to calculate IG score is as follows:

$$\begin{aligned} &IG(\text{feature}) \\ &= -\sum_{k=1}^{m} P(C_k)\ \log P(C_k) \\ &+ P(\text{feature}) \sum_{k=1}^{m} P(C_k|\text{feature})\ \log P(C_k|\text{feature}) \quad (3) \\ &+\ P\left(\overline{\text{feature}}\right) \sum_{k=1}^{m} P\left(C_k\middle|\overline{\text{feature}}\right)\ \log P\left(C_k\middle|\overline{\text{feature}}\right) \end{aligned}$$

where 1<=k<=m and m is the number of classes

$P(C_k)$: Probability of a class C_k

P (feature): Probability of feature

$P(C_k|\text{feature})$: Conditional Probability of class C_k given presence of feature

P ($\overline{\text{feature}}$): Probability of absence of feature

$P\left(C_k\middle|\overline{\text{feature}}\right)$: Conditional Probability of class C_k given absence of feature

The features having higher information gain have more discriminating power.

Standard Deviation [25]: Standard deviation (SD) is a statistical tool that is used to measure deviation of a value from its mean. In feature selection, Sdev calculates the amount of dispersion of a feature from average in the feature space. The higher value of standard deviation shows that the feature is distributed over large range of values thus will be useful in discrimination between classes. For two-class problem of sentiment classification, this value is calculated using the following formula:

$$SD(f_i) = |\text{Sdev}(f_i, C_1) - \text{Sdev}(f_i, C_2)| \quad (4)$$

$$Sdev(f_i, C_k) = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(x_{ji} - mean_k(f_i)\right)^2} \quad (5)$$

where k=1 or 2 for binary classification problem, x_{ij} is the weight of j^{th} feature in the i^{th} sample, mean$_k$ and Sdev(f_i,C_k) is the mean and standard deviation of the i^{th} feature to the kth class.

Gini Index [26], [27]: Gini Index (GI) is the modified version of attribute based feature selection and used as a feature selector for text categorization problems. It is global feature selection method and assign a positive score to each feature. The maximum the score of the feature, the better is its rank.

The GI score is calculated using the following formula:

$$GI(\text{feature}) = \sum_{k=1}^{m} P(\text{feature}|C_k)^2\ P(C_k|\text{feature})^2 \quad (6)$$

where 1<=k<=m and m is the number of classes

P (C_k | feature): Conditional Probability of class C_k given presence of feature

P (feature|C_k): Conditional Probability of feature given presence of class C_k

D. Sentiment Classification Using Supervised Learning

We utilized the following machine learning algorithms for our experimental study as these two algorithms are able to achieve best accuracy in text classification problems.

1) Naïve Bayes classifier (NB)

Naïve Bayes is widely used in text classification as it is computationally efficient and has good classification accuracy. In recent years, Naïve Bayes classifier has been applied in various text classification problems [28, 29]. The probabilistic model uses Bayes theorem to predict the probability of a given feature set belonging to the particular class:

$$P(C_i|\text{features}) = \frac{P(C_i)*P(\text{features}|C_i)}{P(\text{features})} \tag{7}$$

where *P* (C_i) is the prior probability of a class *i*. *P* (C_i | features) is the prior probability of feature set being classified to class i. *P* (features) is the prior probability that feature set has occurred, where *P* (features) is constant for all classes so ignored. The class C_i for which *P* (C_i | features) is maximized determines the class of feature set.

2) Support Vector machine (SVM)

SVM is introduced as a new learning method [30]. It is the fast and accurate method for classification suitable for both linear and non-linear data. SVM is popular in text classification [31], [32] as the classifier has the ability to deal with sparse document vectors created in the case of textual data. Also the method is less prone to overfitting problem and can deal with high dimensional space. The method searches for the maximal marginal hyperplane that separates the two classes using support vectors. These support vectors are the most difficult tuples to classify and carry maximum information. Also support vectors provide compact description of the learned model. A separating hyperplane can be written as follows:

$$W.X + b = 0 \tag{8}$$

where *W* is a weight vector *W*= {*w*1, *w*2…………*wn*} where *n* is the number of features and b is scalar value known as bias. For binary classification problem any tuple that falls above the hyperplane belongs to class +1 and any tuple that falls below or on hyperplane belongs to class -1.

IV. Ensemble of Feature Selection Methods Using HFS (HFS-FS)

In this section, the ensemble of feature selection using hesitant fuzzy set (HFS-FS) is presented. The method consists of two phases: in the first phase, feature relevancy is measured according to five ranking algorithms. In the second phase, the feature importance measured by individual feature rankers is integrated using hesitant fuzzy sets. According to existing literature, decision making problems which involves opinion of different experts on same entity can be modeled using HFS. According to basic principle of HFS, a vector of membership values is needed for generating HFS. In our work, HFS is generated by considering five feature selection methods: Chi-square test, Information gain, Document frequency, and Standard deviation and Gini index. Each method give different relevancy score to every extracted feature in the dataset. The process of feature subset selection is explained below:

Phase 1: Create hesitant fuzzy sets HFSs for n features in the dataset. HFS consist of two parts as shown below:

$$HFS_i = \{\langle F_i, H_k(F_i)\rangle | F_i \in FS; i = 1,2 \ldots n\} \tag{9}$$

where F_i represents i^{th} feature and $H_k(F_i)$ represent its membership value according to relevancy score by k^{th} feature selection method where *k* varies from 1 to 5.The relevancy score of each feature is normalized in range of [0,1] using min-max normalization which is computed as follows:

$$\mu'_k(F_i) = \frac{\mu_k(F_i) - (min(\mu_k))}{(max(\mu_k) - min(\mu_k))} \tag{10}$$

where max (μ_k) and min (μ_k) represent the maximum and minimum score obtained by k^{th} feature selection method.

$$H_k(F_i) = \{\mu'_1(F_i), \mu'_2(F_i), \mu'_3(F_i), \mu'_4(F_i), \mu'_5(F_i)\} \tag{11}$$

where μ_1' (*Fi*) is normalized relevancy score from Chi-square test, μ_2' (*Fi*) is normalized relevancy score generated from Information gain, μ_3' (*Fi*) is normalized relevancy score generated from document frequency, μ_4' (*Fi*) is normalized relevancy score generated from standard deviation and μ_5' (*Fi*) is normalized relevancy score generated from Gini Index.

Phase 2: Compute the overall information energy for HFS.

calculate the overall information energy EHFS of ith feature using hesitant fuzzy set HFSi, the following formula is used:

$$E_{HFS}(i) = \frac{1}{k}\sum_{j=1}^{k}\left(\mu'_j(F_i)\right)^2 \tag{12}$$

where *k* is the number of feature selection methods in our case *k*=5.

The information energy of i^{th} feature computed in Equation 12 is stored at the i^{th} position of a relevancy feature vector RFV of size *n* as follows:

$$RFV(i) = E_{HFS}(i) \tag{13}$$

where 1<=*i*<=*n* and *n* is the number of extracted features from dataset.

The feature F_i are ranked according to weight of feature in RFV (*i*). The higher the value of the feature in RFV, better the rank of the feature.

$$\text{rank}(F_i) \geq \text{rank}(F_j) \text{ if } RFV(i) \geq RFV(j) \tag{14}$$

The subsets of top-k ranked features is selected and evaluated by two different classifiers to obtain best feature subset with maximum relevancy. The ensemble of feature rankers using hesitant fuzzy set is explained in Algorithm 1 and 2:

Algorithm 1: Feature ranking using hesitant fuzzy sets (FR-HFS)

Input: Dataset D having m-unprocessed reviews and k-features to be selected
Output: Best ranked k-features

```
1.  Extract n-features based on Unigram or POS based feature
    representation method
2.  Create a document-feature vector DFV of size $m$ x $n$ using TF-
    IDF as a weighing method
3.  FS={1,2…………$n$}//Store all features in Feature Subset FS
  // Create HFS for all features using 5-feature
    ranking methods
4.  for $i$ ← 1: $n$
      Create $HFS_i$ using Equation 9
5.  endfor
// Compute information energy of all features
   using integration of all feature selection methods
6.  for $i$ ← 1: $n$
    Compute $E_{HFS}$ using Equation 12
7.  endfor
// Store overall relevancy score of each feature in
   RFV vector of size $n$
8.  for $i$ ← 1: $n$
    Store the $E_{HFS}$ of each feature in RFV
    using Equation 13
9.  endfor
// Feature sorting using RFV
10. Sort FS according to RFV
11. Return FS[1:k]: best ranked k-features from FS
```

In Algorithm 1, dataset is first preprocessed and Unigram or POS features are extracted from m-samples of dataset D. In the second step, the term-document matrix is generated using term-frequency and inverse document frequency (tf-idf) of extracted features. The hesitant fuzzy set is then created for each feature using five feature selection methods as discussed in the previous section. The relevancy score of each feature is obtained and stored in the relevancy feature vector (RFV). The top-k features are the features that have best relevancy score in RFV.

Algorithm 2: Feature Subset Selection Method (FSS)

Input: Classifier C ←{C_1, C_2…C_n} and Datasets D ←{d_1, d_2…d_m}
Range (k1, k2) and variation *var*
Output: Subset size and accuracy

```
1.  Initialize $k$1=0.05*size(FS) and $k$2=.3*size(FS)
2.  Auc1=[ ]
3.  Auc2=[ ]
4.  $k$=$k$1
5.  for $d_i$ ε $D$ do:
6.    for $C_j$ ε $C$ do:
7.      while $k$<=$k$2:
          i.   P1=predict_accuracy ($C_j$,FR-
               HFS($d_i$ ,$k$)) // 5-fold cross validation
          ii.  Auc1.add(P1)
          iii. Auc2.add(k)
          iv.  $k$=$k$+var
8.      best_accuracy($d_i$,$C_j$)=max(Auc1)//best
        accuracy obtained on dataset $d_i$ by
        classifier $C_j$
9.      ind($d_i$,$C_j$)=index(max(Auc1))
10.     subset_size($d_i$,$C_j$)=Auc2[ind]// subset
      size giving best accuracy
11.  for all $i$ ε{1,2..$m$} and j ε {1,2…$n$}
12.    Return best_accuracy($d_i$,$C_j$): Best
       accuracy and subset_size($d_i$,$C_j$):subset
       size
```

The best accuracy score and subset size is obtained on different datasets using both NB and SVM classifier as shown in Algorithm 2. To predict the real performance of classifier in terms of accuracy, 5-fold cross validation is employed on dataset considering top-k features returned by Algorithm 1 where k ranges from 5-30% of total extracted features. The value of k that results in best accuracy score is returned as the final subset size for that classifier and dataset.

V. Experiments and Results

The performance of the proposed algorithm is evaluated on Naïve Bayes and Support Vector machine classification algorithms using five-fold cross validation. In this section, we discuss the performance metrics, datasets, validation techniques, performance measure and finally result analysis.

A. Performance Metrics

To evaluate classifier performance, we used Accuracy, Precision and Recall in our work [33]. Accuracy is the percentage of test tuples that are correctly labelled by the classifier. Accuracy cannot be the only evaluation measure as it is possible that a system showing 90% accuracy may be poor system because it is identifying only one class correctly not the other in binary classification. For an accurate system, negative as well as positive tuples should be correctly classified. Precision and recall are used to check accuracy with respect to both positive and negative tuples. These measures are calculated as follows:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{15}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{16}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+TN+FP} \tag{17}$$

Here *TP* is the positive items that are correctly classified as positive, *TN* is the negative items correctly classified by as negative, *FP* is the negative tuples incorrectly classified as positive and *FN* is the positive items incorrectly classified as negative.

B. Cross Validation Technique

We used five-fold cross validation technique to get reliable and unbiased results of the classifier. In this technique, dataset is divided into 5 folds: out of which four folds are kept for training and one fold for testing. In each run, test set is varied and performance is evaluated by the classifier. This process is repeated for 5 times and mean of the predicted results is computed to get final performance of the classifier.

TABLE I: Review Datasets Statistics

Dataset	No. of positive samples	No. of negative samples	Feature Representation	Total no. of extracted features
Movie	1000	1000	Unigrams	39389
			POS	178356
Book	1000	1000	Unigrams	15123
			POS	28033
Music	1000	1000	Unigrams	13632
			POS	25798

C. Datasets

The datasets used for conducting experimental work are the most widely used datasets for sentiment analysis: Movie Review dataset [34] and Amazon product reviews [35]. The amazon product reviews dataset consist of reviews of 25 different products. We used reviews of only two products: Book and Music from amazon dataset. The datasets used in our study is balanced data containing equal number of positive and negative samples. The dataset statistics are shown in Table I.

D. Preprocessing

To extract unigram and POS features from text reviews as shown in the work [18], we adopted two different methods of preprocessing. In the first step, each review or document is converted to lower case and then tokenized into sentences. In the next step, each sentence is tokenized to words to extract unigram features. In this process, all stop words and words having document frequency less than 5 are eliminated. The term-document matrix is constructed using popular TF-IDF feature weighing scheme after feature extraction from review.

TABLE II: POS-PATTERNS FOR FEATURE EXTRACTION AND POS FEATURE EXTRACTED ON MUSIC DATASET

Feature Class	POS-Patterns	POS feature with tag
Noun	N,NN,NNN,NI,NNI,NIN,NV	('rap music today', NN NN NN) ('creativity', NN)
Adjective	J,JN,JJN,JNN,JNI	('bad performer', JJ NN) ('modern day version', JJ NN NN)
Verb	VI,VNN,VN,VNI,VV,VVI,VJN	('was amazed at' ,VBD VBN IN) ('relax children', VB NNS)
Adverb	R,RR,RRJ,RRR	('hopefully', RB) ('also very nice', RB RB JJ)

To extract POS features from text, word on each tokenized sentence is further tagged by NLTK POS tagger. The POS patterns of length < 3 are used as linguistic filters to annotate the reviews using POS tags. The four class of pattern can represent a feature in sentiment analysis problems. The class of feature and linguistic filters used to extract that feature is shown in Table II. The last column of the Table II shows few POS features extracted from Music Review datasets using POS patterns.

Table III shows the feature names and relevancy score of some of the best ranked top-100 POS features from Book dataset. The top ranked feature 'book' has relevancy score 0.601.

TABLE III: RELEVANCY SCORE OF BEST FEATURES ON BOOK DATASET

Feature name	Relevancy score	Feature rank
boring	0.4021	2
bestseller	0.1916	7
waste of time	0.1117	23
excellent book	0.11742	18
good thriller	0.0733	80
not enough	0.07086	97
great condition	0.0693	100

POS feature are ranked on the basis of their relevancy score and then top-k features are selected for evaluating performance of the classifiers.

E. Experimental Settings

We conduct the experiments on Lenovo Ideapad 310 with 2.5 GHz Intel Core I5 processor and 8GB RAM. Anaconda with Python 3.4 is used to design our project. For the initial steps of preprocessing and feature extraction, NLTK is utilized. It is a tool in Python with a set of text processing libraries for classification, tokenization, stemming, tagging, parsing etc. Scikit-learn (sklearn) in Python is used for sentiment classification. It is simple and efficient tool in python used for data mining and data analysis tasks and the tool is built on Numpy, Scipy and Matplotlib in Python. For data analysis and graph plotting, Matplotlib library of Python is utilized.

F. Performance Evaluation

We evaluated the performance of our proposed system on three review datasets: Movie, Book and Music using two feature representation schemes: Unigram and POS features. We employed Naïve Bayes and Support Vector machine for classification as they are the most popular algorithms in the area of text classification. The results of accuracy, precision and recall of NB and SVM classifiers on three datasets using 5-fold cross validation techniques has been presented in this section.

Table IV shows the classifier performance using accuracy, precision and recall on all three datasets without feature selection. The results are obtained for both feature representation schemes. We can depict from Table IV that SVM outperforms NB in terms of accuracy and precision in case of Movie and Book review dataset represented by both unigram and POS features whereas NB outperforms SVM in terms of recall for the same.

TABLE IV: ACCURACY OF CLASSIFIERS ON UNIGRAM AND POS FEATURES USING 5*5 CROSS VALIDATION WITHOUT FEATURE SELECTION

		Unigram			POS		
	Classifier	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
Movie	SVM	84.8	85.9	83.3	85.1	85.7	84.3
	NB	81.64	79.4	85.5	82.6	81.7	84.2
	Avg	83.22	82.6	84.4	83.8	83.7	84.2
Book	SVM	77.1	78.5	74.6	76.2	77.3	74.3
	NB	76.45	72.8	84.6	76.3	73.5	82.2
	Avg	76.78	75.6	79.6	76.3	75.4	78.2
Music	SVM	75.45	74.7	76.8	75.7	75.8	75.7
	NB	76.5	77.1	75.8	76.8	76.5	77.7
	Avg	75.98	75.9	76.3	76.2	76.1	76.7

The Fig. 1-3 shows the detailed analysis in terms of precision, recall and accuracy on all three datasets with varying size of feature subset. The size of feature subset is chosen based on the number of total extracted features using both types of feature representation. The analysis is done by choosing 5-30% of best ranked features termed as 'k' from the proposed HFS-FS.

Fig. 1(a) and Fig. 1(b) depicts the classifiers performance on unigrams features of Movie Reviews. The value of *k* is chosen in the range of 1000-5000 with a variation of 200. As shown in the figure, both classifiers give best accuracy value of around 90% at *k*=2400 but the precision value is best at

2400 for SVM classifier and 2000 for NB classifier. The performance graph in Fig. 1(c) and Fig. 1(d) depicts best performance of classifier on selecting around 20000 feature with an average accuracy score of 90%. The value of *k* is chosen in the range of 5000-25000 with a variation of 1000.

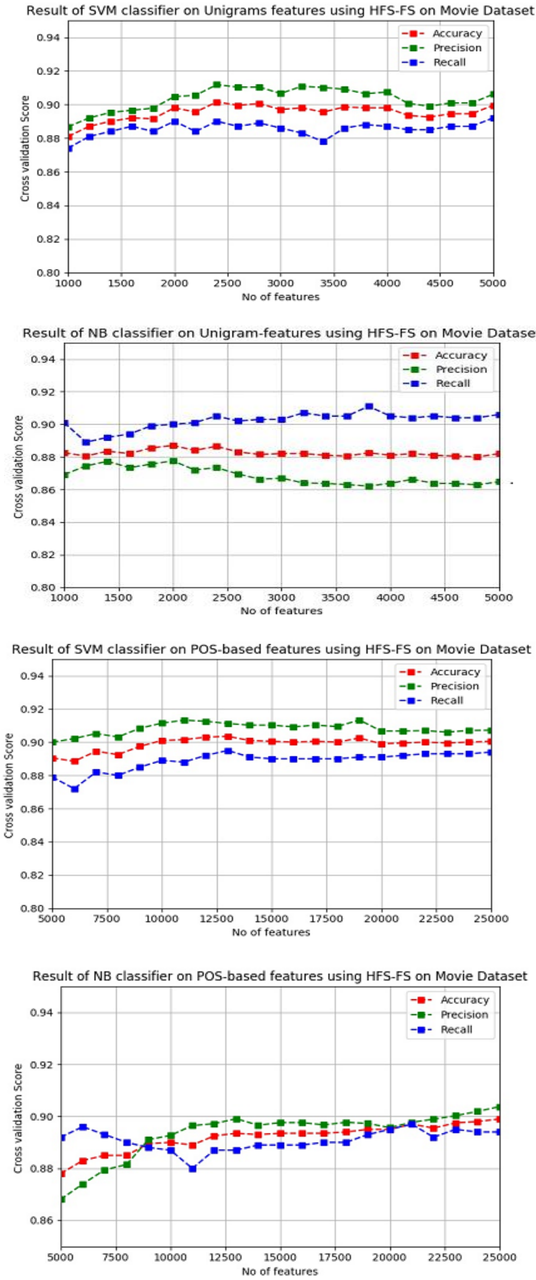


Fig. 1. (a)-(d) performance analysis on movie dataset.

Fig. 2(a) and Fig. 2(b) depicts classifiers performance on unigrams features of Book Reviews. The value of *k* is chosen in the range of 1000-5000 with a variation of 200. As shown in the figure, NB classifier outperforms SVM and give best accuracy value of around 88.35% and precision value as 88.5% at *k*=1000. The performance graph in Fig. 2(c) and Fig. 2(d) depicts that classifier performance on POS features where value of *k* is 1000-10000 with variation of 500. In this case also, NB outperforms SVM and gives an accuracy of 87.5% on selecting only 6500 informative features (around 23%) from full feature set size of 28033.The graph also shows that only small feature size of 1000 from extracted unigrams can give good results on Book reviews using our proposed method.

Fig. 3(a) and Fig. 3(b) depicts classifiers performance on unigrams features of Music Reviews. The value of k is chosen in the range of 1000-5000 with a variation of 200. NB classifier outperforms SVM and give best accuracy value of around 87.6% and precision value as 88.5% at *k*=1800. The performance graph in Fig 3 (c) and (d) depicts the classifier performance on POS features where value of k is in the range of 1000-6000 with variation of 200. In this case also, NB outperforms SVM and gives an accuracy of 86.39% on selecting only 2800 informative features (around 11%) from full feature set size of 25798.The graph also shows that only small feature size from extracted POS features can give good results on Music reviews using our proposed method.

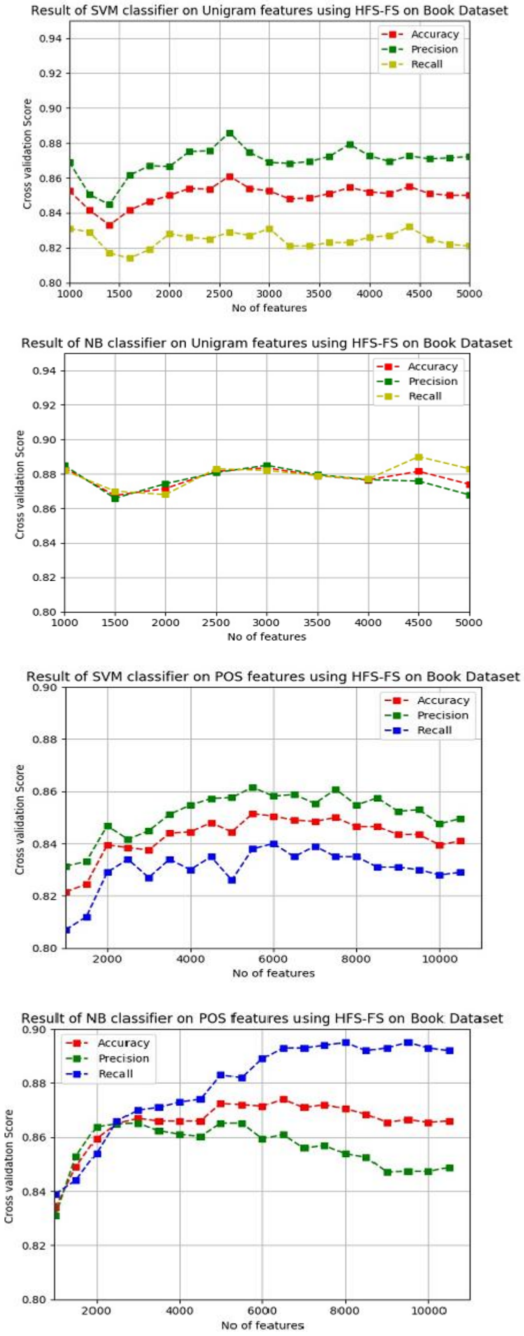
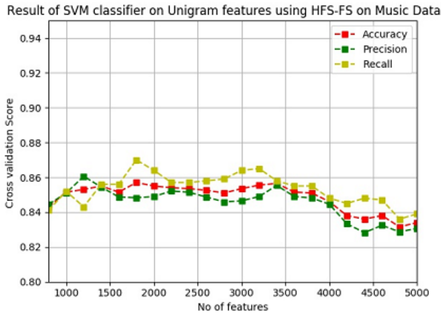


Fig. 2. (a)-(d) Performance analysis on book dataset.

Table V shows the comparison between NB and SVM classification algorithms without feature selection and with HFS-FS. The average accuracy score predicted by both the classifiers on Unigram and POS features achieves a significant rise of around 6% using HFS-FS on Movie Reviews. The accuracy score by both classifiers on Unigram and POS features using our proposed feature selectors improved the accuracy by around 10% on Book reviews. Music reviews dataset also shows tremendous improvement of around 10% in accuracy value on both type of features using HFS-FS.

Based on the results obtained, Naïve Bayes performs better than SVM on all datasets except Movie Reviews. Also our proposed method gives better accuracy when HFS-FS selects from Unigram features in case of Book and Music dataset whereas Movie reviews dataset achieves better performance with POS features. The results also indicate reduction in dimension of unigram features and POS features to 14% on an average across all three datasets.



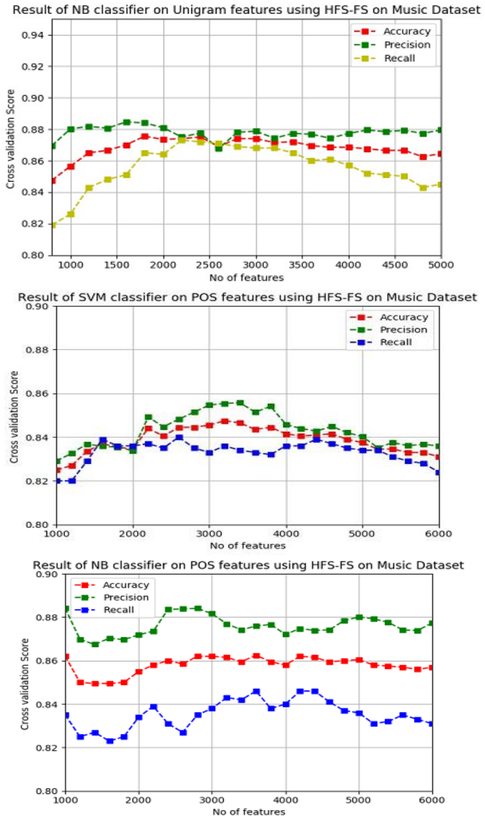


Fig. 3. (a)-(d) Performance analysis on music dataset.

TABLE V: PERFORMANCE ANALYSIS IN TERMS OF ACCURACY BETWEEN FULL AND BEST FEATURE SET ON THREE DATASETS

		Unigram-features				**POS-features**			
Dataset	**Classifier**	**# of features**	**Accuracy**	**# of features subset**	**Accuracy**	**# of features**	**Accuracy**	**# of features subset**	**Accuracy**
Movie	SVM	39389	0.848	2400	90.14	178356	85.1	18000	90.35
	NB		0.8164	2400	88.88		82.65	20000	89.8
	Average		0.8322	2400	**89.51**		83.875	19000	**90.075**
Book	SVM	15123	77.1	2600	86.01	28033	76.25	5500	85.2
	NB		76.45	1000	88.35		76.3	6500	87.5
	Average		76.775	1800	**87.18**		76.275	6000	**86.35**
Music	SVM	13632	75.45	1800	85.7	25798	75.7	3200	84.79
	NB		76.5	1800	87.6		76.8	2800	86.39
	Average		75.975	1800	**86.65**		76.25	3000	**85.59**

TABLE VI: RESULT OF PAIRED T-TEST ON BASELINE AND PROPOSED METHOD

							Paired T-test	
Dataset	**Classifier**	**Feature Representation**	**Baseline**	**#of subset**	**Baseline Accuracy**	**Ensemble Accuracy**	**t-value**	**H-value**
Movie	SVM	Unigram	CHI	3000	91.8	90.125	2.07E+00	0
			IG	3000	84.115	90.125	-7.78E+00	1
			GI	3000	85.25	90.125	-4.87E+00	1
	SVM	POS	CHI	20000	73.65	89.9	-1.09E+01	1
			IG	20000	86.25	89.9	6.30E-02	0
			GI	20000	89.7	89.9	2.75E+00	0
Book	NB	Unigram	CHI	3000	88.8	88.35	5.53E-01	0
			IG	3000	77.35	88.35	-1.50E+01	1
			GI	3000	79.45	88.35	-9.56E+00	1
	NB	POS	CHI	4500	64.85	86.6	-2.62E+01	1
			IG	4500	77.55	86.6	-1.20E+01	1
			GI	4500	79.6	86.6	-1.85E+01	1
Music	NB	Unigram	CHI	2800	88.35	87.4	1.05E+00	0
			IG	2800	77.2	87.4	-1.60E+01	1
			GI	2800	79.45	87.4	-1.04E+01	1
			CHI	2800	60	86.39	-3.05E+01	1
			IG	2800	76.95	86.39	-2.43E+01	1
	NB	POS	GI	2800	78.55	86.39	-1.29E+01	1

G. Discussions

The proposed ensemble of filter-based feature selectors is independent of the individual classifier. The selection of features is based on the relevancy score of the features which is computed by integration of different filter based feature selection methods using hesitant fuzzy sets. The low computational cost of the algorithm makes it more suitable for large datasets. The ensemble of feature selectors benefit the weak filter-based feature selection methods without compromising the cost.

TABLE VII: COMPARISON OF BASELINE INTEGRATION METHODS AND PROPOSED METHOD IN TERMS OF ACCURACY ON ALL THREE DATASETS

	Unigram-based				**POS-based**			
Dataset	**OIFV [18]**	**FIFS [18]**	**Word-elation [16]**	**HFS-FS**	**OIFV [18]**	**FIFS [18]**	**Word-relation [16]**	**HFS-FS**
Movie	90.7	**90.8**	87.7	89.5	92.4	**92.9**	86.8	90.1
Book	84.7	85.1	81.8	**87.1**	84.6	84.1	80.1	**86.3**
Music	84.6	85.6	-	**86.6**	83.3	84.1	-	**85.6**

To prove that there is significant difference between the proposed approach and baseline approaches of feature selection methods, the paired t-test [36] is conducted. Table VI shows the results of statistical paired t-test based on accuracy score obtained from 5-fold cross validation method using HFS-FS or baseline algorithms for feature selection. We choose Chi-square, Information gain and Gini index as baseline feature rankers and evaluated their performance using any one of the classifier. The t-value indicate paired t-test value for comparing the means of two methods. Based on this value and 5% confidence level, the hypothesis H is rejected or accepted. H=0 indicates not statistically different and H=1 indicates statistically significant difference between the baseline and proposed method. The results on all three datasets shown in Table 6 depicts that HFS-FS outperform all baseline feature selection methods except Chi-square test when classifier performance is predicted on unigram features. The POS-based features when selected using HFS-FS shows great improvement in accuracy than all baseline algorithms and statistical test value indicates this difference in accuracy to be statistically significant on all datasets.

Finally, our proposed approach is compared with the work [16], [18] in terms of accuracy measure as performance metrics on all three datasets using both type of feature representation methods. The comparison is done to conclude whether our approach outperforms the integration methods proposed in the past. Table VII shows the comparison between our approach and baseline integration approaches used in the area of sentiment analysis. It may be noted that results on Music dataset is not available in the work [16].

The results indicate that HFS-FS outperform baseline integration algorithms in terms of accuracy in case of Book and Music datasets. It shows around 2% improvement in average accuracy score when best ranked k-features are selected using our proposed ensemble of feature selection methods. However, on Movie review dataset, difference is not so significant.

REFERENCES

[1] 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.

[2] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. the 2002 Conference on Empirical Methods in Natural Language Processing*, 2002, pp. 79-86.

[3] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computer & Electrical Engineering*, vol. 40, no. 1, pp. 16-28, 2014.

[4] P. D. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," in *Proc. 40th Annual Meeting on Association for Computational Linguistics*, 2002, pp. 417-424.

[5] H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10760-10773, 2009.

[6] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of Sentimental Reviews Using Machine Learning Techniques," *Procedia Computer Science*, vol. 57, pp. 821-829, 2015.

[7] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Systems with Applications*, vol. 57, pp. 117-126, 2016.

[8] A. Abbasi, H. Chen, A. Salem, A. Abbasi, H. Chen, and A. Salem, "Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums," *ACM Transactions on Information Systems*, vol. 26, no. 3, pp. 1-34, 2008.

[9] T. O'Keefe and I. Koprinska, "Feature selection and weighting methods in sentiment analysis," *Australas. Doc. Comput. Symp. ADCS*, pp. 67-74, 2009.

[10] M. Rogati and Y. Yang, "High-performing feature selection for text classification," in *Proc. the Eleventh International Conference on Information and Knowledge Management*, 2002, pp. 659-661.

[11] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with Naïve Bayes," *Expert System with Applications*, vol. 36, no. 3, pp. 5432-5435, 2009.

[12] T. Parlar, S. A. Özel, and F. Song, "QER: A new feature selection method for sentiment analysis," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1-19, 2018.

[13] K. K. Bharti and P. K. Singh, "Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering," *Expert Systems with Applications*, vol. 42, no. 6, pp. 3105-3114, 2015.

[14] H. H. Hsu, C. W. Hsieh, and M. D. Lu, "Hybrid feature selection by combining filters and wrappers," *Expert System with Applications*, vol. 38, no. 7, pp. 8144-8150, 2011.

[15] Y. Saeys, T. Abeel, and Y. Van de Peer, "Robust feature selection using machine learning and knowledge discovery in databases," *Lecture Notes in Computer Science*, vol. 5212, pp. 313-325, 2008.

[16] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Information Sciences*, vol. 181, no. 6, pp. 1138-1152, 2011.

[17] A. Onan, S. Koruko, and S. Glu, "A feature selection model based on genetic rank aggregation for text sentiment classification," *Journal of Information Science*, vol. 43, no. 1, pp. 25-38, 2017.

[18] A. Yousefpour, R. Ibrahim, and H. N. A. Hamed, "Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis," *Expert Systems with Applications*, vol. 75, pp. 80-93, 2017.

[19] V. Torra and Y. Narukawa, "On hesitant fuzzy sets and decision," in *Proc. IEEE International Conference Fuzzy Systems*, 2009, pp. 1378-1382.

[20] V. Torra, "Hesitant fuzzy sets," *International Journal of Intelligent Systems*, vol. 25, no. 6, pp. 529-539, 2010.

[21] G. Qian, H. Wang, and X. Feng, "Generalized hesitant fuzzy sets and their application in decision support system," *Knowledge-Based Systems*, vol. 37, pp. 357-365, 2013.

[22] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," *Mining Text Data*, Springer, Boston, MA, pp. 415-463, 2012.

[23] A. Kennedy, I. Diana, and D. Inkpen, "Sentiment classification of movie and product reviews using contextual valance shifters," *Computational Intelligence*, vol. 22, no. 2, pp. 110-125, 2006.

[24] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Fourteenth International Conference of Machine Learning*, 1997, pp. 412-420.

[25] A. Yousefpour, R. Ibrahim, H. N. A. Hamed, and M. S. Hajmohammadi, "Feature reduction using standard deviation with different subsets selection in sentiment analysis," in *Proc. Asian Conference on Intelligent Information and Database Systems*, 2014, pp. 33-41.

[26] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, vol. 3, pp. 1289-1305, 2003.

[27] W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," *Expert Systems with Applications*, vol. 33, no. 1, pp. 1-5, 2007.

[28] E. Frank and R. R. Bouckaert, "Naive bayes for text classification with unbalanced classes," *Knowledge Discovery in Databases: PKDD 2006*, pp. 503-510, 2006.

[29] S. B. Kim, K. S. Han, H. C. R. Rim, and S. H. Myaeng, "Some effective techniques for Naïve bayes text classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 11, pp. 1457-1466, 2006.

[30] C. Cortes and V. Vapnik, "Support vector machine," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[31] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proc. European Conference on Machine Learning*, 1998, pp. 137-142.

[32] S. Tong and D. Koller, *Support Vector Machine Active Learning with Applications to Text Classification*, vol. 2, no. 11, pp. 45-66, 2001.

[33] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, Elsevier, 2006.

[34] B. Pang and L. Lee, "A sentimental education," in *Proc. the 42nd Annual Meeting on Association for Computational Linguistics*, 2004, pp. 271-277.

[35] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," *ACL*, vol. 45, no. 1, pp. 440.-447, 2007.

[36] D. R. Cooper and P. S. Schindler, *Business Research Methods*, Mc Graw Hill, 2003.

Gunjan Ansari is a research scholar in the Department of Computer Engineering, Jamia Millia Islamia, Delhi. She received her B.E. (CSE) and M.Tech. (CSE) degree from Rajiv Gandhi Technical University, Bhopal. Her areas of interest are in data mining, natural language processing and algorithms.

Tanvir Ahmad is a professor and head of the Department of Computer Engineering, Jamia Millia Islamia, Delhi. He received his Ph.D. on the topic "Frequent and sequential pattern mining and their applications" from Jamia Millia Islamia. His areas of interests are in text mining, graph mining, big data analytics, natural language processing and information security

Mohammad Najmud Doja was the founder head of the Department of Computer Engineering and Director, CIT at Jamia Millia Islamia University, New Delhi. Currently, Prof. Doja is the director of Indian Institute of Information Technology (IIIT), Sonepat. He has published more than 200 papers in international journals and conferences. He has also guided more than 24 PhDs in the area of networking, soft computing, opinion mining etc.