FEATURE SELECTION AND CLASSIFICATION APPROACH FOR SENTIMENT ANALYSIS

Gautami Tripathi[1] and Naganna S.[2]

[1]PG Scholar, School of Computing Science and Engineering,
Galgotias University, Greater Noida, Uttar Pradesh.
[2]Assistant Professor, School of Computing Science and Engineering,
Galgotias University, Greater Noida, Uttar Pradesh.

ABSTRACT

Sentiment analysis and Opinion mining has emerged as a popular and efficient technique for information retrieval and web data analysis. The exponential growth of the user generated content has opened new horizons for research in the field of sentiment analysis. This paper proposes a model for sentiment analysis of movie reviews using a combination of natural language processing and machine learning approaches. Firstly, different data pre-processing schemes are applied on the dataset. Secondly, the behaviour of two classifiers, Naive Bayes and SVM, is investigated in combination with different feature selection schemes to obtain the results for sentiment analysis. Thirdly, the proposed model for sentiment analysis is extended to obtain the results for higher order n-grams.

KEYWORDS

Sentiment Analysis; Opinion Mining; Information Retrieval; Web Data Analysis; Feature Selection; User Generated Content; Pre-Processing.

1. INTRODUCTION

The evolution of web technology has led to a huge amount of user generated content and has significantly changed the way we manage, organize and interact with information. Due to the large amount of user opinions, reviews, comments, feedbacks and suggestions it is essential to explore, analyze and organize the content for efficient decision making. In the past years sentiment analysis has emerged as one of the popular techniques for information retrieval and web data analysis. Sentiment analysis, also known as opinion mining is a subfield of Natural Language Processing (NLP) and Computational Linguistics (CL) that defines the area that studies and analyzes people's opinions, reviews and sentiments.

Bing Liu [1] defines an opinion as a quintuple $<o_i, f_{ij}, so_{ijkl}, h_i, t_l>$, where o_i is the target object, f_{ij} is the feature of the target object o_i, h_i is the opinion holder, t_l is the time when the opinion is expressed and so_{ijkl} is the sentiment value of the opinion expressed by the opinion holder h_i about the object o_i at time t_l.

Sentiment analysis defines a process of extracting, identifying, analyzing and characterizing the sentiments or opinions in the form of textual information using machine learning, NLP or statistics. A basic sentiment analysis system performs three major tasks for a given document. Firstly it identifies the sentiment expressing part in the document. Secondly, it identifies the sentiment holder and the entity about which the sentiment is expressed. Finally, it identifies the polarity (semantic orientation) of the sentiments. Bing Liu [1] defines opinion mining as the field of study that analyzes people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes.

Sentiment analysis can be performed at three different levels: document, sentence and aspect level. The document level sentiment analysis aims at classifying the entire document as positive or negative, (Pang et al, [2]; Turney, [3]). The sentence level sentiment analysis is closely related to subjectivity analysis. At this level each sentence is analyzed and its opinion is determined as positive, negative or neutral, (Riloff et al, [4]; Terveen et al, [5]). The aspect level sentiment analysis aims at identifying the target of the opinion. The basis of this approach is that every opinion has a target and an opinion without a target is of limited use, (Hu and Liu, [6]).

Today, many companies are using sentiment analysis as the basis for developing their marketing strategies [23] [24][30]. They access, analyze and predict the public opinion about their brand. Researchers are also focusing on developing automatic tools for opinion mining. Several tools are already available in the market that helps companies to extract information from the internet. Some of these tools includes: SenticNet, Converseon, Factiva, Sentiment140 and SocialMention.

In this paper we explored the machine learning classification approaches with different feature selection schemes to obtain a sentiment analysis model for the movie review dataset. Experiments are performed using various feature selection schemes and the results obtained are compared to identify the best possible approach. A pre-processing model for the dataset is also proposed. In the course of this work many previous works are reviewed and some of them are applied in the proposed work.

The proposed work is evaluated by running experiments with the polaritydatasetV2.0, available at http://www.cs.cornell.edu/people/pabo/movie-review-data. Natural Language Processing and Machine Learning approaches were used for the process. Multiple experiments were carried out using different feature sets and parameters to obtain maximum accuracy.

In the final phase of this work the results are evaluated to find the issues, improvements and ways to extend the work. A summary of the obtained results and future scope is also discussed. The results obtained are compared to the previous works to obtain a comparative summary of the existing work and the proposed work.

2. RELATED WORK

The researches in the field of sentiment analysis started much earlier in 1990's but the terms sentiment analysis and opinion mining were first introduced in the year 2003, (Nasukawa et al, [7]; Dave et al, [8]). The earlier work in the field was limited to subjectivity detection, interpretation of metaphors and sentiment adjectives [31][32]. J.M. Wiebe [9] presents an algorithm to identify the subjective characters in fictional narrative text based on the regularities

in the text. M.A. Hearst [10] defines a direction based text interpretation approach for text based intelligent systems to refine the information access task. J.M. Wiebe [11] performed extensive examination to study the naturally occurring narratives and regularities in the writings of authors and presents an algorithm that tracks the point of view on the basis of these regularities.

Hatzivassiloglou and McKeown [12] proposed a method to find the semantic orientation of the adjectives and predicted whether two conjoined adjectives are of same polarity with 82% accuracy. They used a three step process to determine the orientation of the adjectives by analyzing their conjunctions: (1).conjunctions of adjective are extracted from documents. (2).The set of extracted conjunctions are split into test set and training set. The trained classifier is then applied to the test set to produce a graph showing same or different orientation links between the pair of adjectives conjoined in the test set. (3).The adjectives from step2 are partitioned into two clusters. Assuming that the positive adjectives are more frequently used the cluster with higher average frequency is considered to contain positive terms.

L. Terveen et al [5] designed an experimental system, PHOAKS (people helping one another know stuff), to help users locate information on the web. The system uses a collaborative filtering approach to recognize and reuse recommendations. J. Tatemura [13] developed a browsing method using virtual reviewers for the collaborative exploration of movie reviews from various viewpoints. Morinaga et al [14] worked in the area of marketing and customer relationship management and presented a framework for mining product reputation on internet. The defined approach automatically collects the user's opinions from the web and applies text mining techniques to obtain the reputation of the products.

P.D. Turney [3] presents an unsupervised method to classify the reviews as thumbs up (recommended) or thumbs down (not recommended). It uses document level sentiment classification and Pointwise Mutual Information (PMI) to obtain the average semantic orientation of the reviews. The algorithm achieves an average accuracy of 74% for 410 reviews. Later Turney and Littman [15] expanded the work by presenting an approach to find out the semantic orientation of a text by calculating its statistical association with a set of positive and negative words using PMI and Latent Semantic Analysis (LSA). The method when tested with 3596 words (1614 positive and 1984 negative) achieves an accuracy of 82.8%.

Pang et al [2] performed document level sentiment classification using standard machine learning techniques. They used Naïve Bayes, Maximum Entropy and SVM techniques to obtain the results for unigrams and bigrams and was able to achieve 82.9% accuracy using three fold cross validation for unigrams. Their work also focuses on better understanding of the difficulties in the sentiment classification task. Dave et al [8] trained a classifier using reviews from major websites. The results obtained showed that higher order grams can give better results than unigrams.

Esuli and Sebastiani [16] presented an approach to determine the orientation of a term based on the classification of its glosses i.e. the definitions from the online dictionaries. The process was carried out in the following steps, (1). A seed set representing the positive and negative categories is provided as the input. (2). Lexical relations from the online dictionary are used to find new words representing the two categories thus forming the training set. (3). Textual representation of the terms is generated by collating all the glosses of the term. (4). A binary classifier is trained on the training set and then applied to the test set.

Hu et al [17] derives an analytical model to examine whether the online review data reveals the true quality of the product. They analyzed the reviews from amazon. The results showed that 53% reviews had a bimodal and non-normal distribution. Such reviews cannot be evaluated with the average score and thus a model was derived to explain when the mean can serve as the valid representation of a products true quality. It also discusses the implications of this model on marketing strategies.

Ding et al [18] proposed a holistic approach to infer the semantic orientation of an opinion word based on review context and combine multiple opinions words in same sentence. The proposed approach also takes into account the implicit opinions and handles implicit features represented by feature indicators. A system named Opinion Observer was also implemented based on the proposed technique. Murthy G. and Bing Liu [19] proposed a method which study sentiments in comparative sentences and also deals with context based sentiments by exploiting external information available on the web. V Suresh et al [20] presents an approach that uses the stopwords and the gaps between stopwords as the features for sentiment classification.

M. Rushdi et al [21] explored the sentiment analysis task by applying SVM for testing different domains of dataset using several weighing schemes. They used three corpora for their experimentation including a new corpus introduced by them and performed 3-fold and 10-fold cross validations for each corpus.

The last two decades have seen significant improvement in the area of sentiment analysis or opinion mining. A number of research papers have also been published presenting new techniques and novel ideas to perform sentiment analysis [26][27][28][29][33]. Still there is not much work in the field of data extraction and corpus creation. From the discussions made in the previous paragraphs it has been observed that most of the work in this field focuses on finding the sentiment orientation of the data at various levels but very few uses data pre-processing and feature selection as the basis for accuracy improvement. The other observation is that almost all approaches used the lower order n-grams (unigrams and bigrams) for experimentation. The work by Pang et al [2][25] mention of unigrams and bigrams only. Later Dave et al [8] extended the work to trigrams. N-grams of order higher than three (trigrams) have not been explored to considerable levels. By considering the above observations as the research gaps we made a problem statement and proposed a methodology in the next section. Our proposed method focuses on efficient data pre-processing and compare various feature selection schemes and extends the results for higher order n-grams (trigrams and 4-grams).

3. PROBLEM STATEMENT AND PROPOSED TECHNIQUE

This section presents the proposed technique to analyze sentiments in a movie domain. The proposed approach uses a combination of NLP techniques and supervised learning. In the first stage a pre-processing model is proposed to optimize the dataset. In the second stage experiments are performed using the machine learning methods to obtain the performance vector for various feature selection schemes. We used up to 4-grams (i.e. n=1, 2, 3, 4) in this work. The model for the proposed technique is depicted in figure 1.

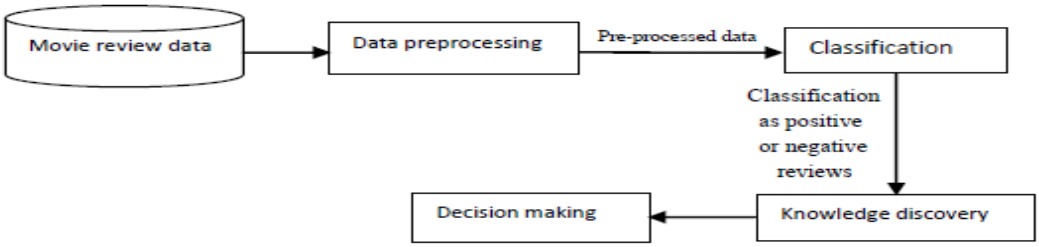


Figure1. Proposed Framework for Sentiment Analysis.

3.1 Experimental setup

We have used Rapid Miner Studio 6.0 software with the text processing extension, licensed under AGPL version3, and Java1.6. Rapid Miner supports the design and documentation of overall data mining process. We have implemented our model using the Linear Support Vector Machine learner that uses the java implementation of SVM, *mySVM*by Stefan Ruping. Firstly, we pre-process the training dataset (polaritydatasetV2.0) and then using 5-fold cross-validation we train the Linear SVM classifier. Tests were also conducted using the Naïve Bayes classifier and various feature selection schemes.

3.2 Data pre-processing

The general techniques for data collection from the web are loosely controlled and therefore the resultant datasets consist of irrelevant and redundant information. Several pre-processing steps are applied on the available dataset to optimize it for further experimentations. The proposed model for data pre-processing and the corresponding algorithm is shown in figure2 and figure3 respectively.

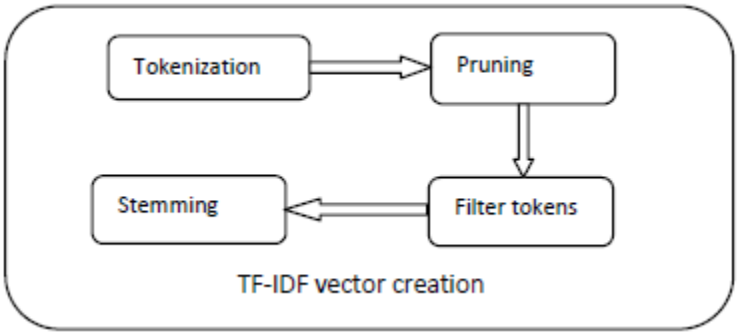


Figure2. Proposed Model for Data Pre-processing

1. *Input dataset, d.*
2. *Read text from dataset to generate new dataset d_1.*
2. *Construct a tokenset, T for the dataset d_1.*
3. *For each token t_i in the set T do*

   *If(5<= ft_i <=1990) /*total number of documents in which the term i occurs. */*
   New token set $T_1 \leftarrow t_i$
   Else
   Discard token t_i

4. *For each token t_j in the new token set T_1 do*

   If (4<= token_length<= 25)
   New token set $T_2 \leftarrow t_j$
   Else
   Discard token t_j

5. *apply stemming algorithm to the new token set T_2*

   Stemming language (T_2) ← English.

Figure3. Algorithm to Create Word Vector from Text Collection Stored in Multiple Files.

Tokenization

This process splits the text of a document into sequence of tokens. The splitting points are defined using all non letter characters. This results in tokens consisting of one single word (unigrams).

Pruning

The movie review data set was pruned to ignore the too frequent and too infrequent words. Absolute pruning scheme was used for the task. Two parameters were used for the pruning task namely, *prune below* and *prune above*. The value of these parameters was set as: *pruned below =5 and pruned above =1990* i.e. ignoring the words that appear in less than 5 documents and in more than 1990 documents.

Filtering tokens

Length based filtration scheme was applied for reducing the generated token set. The parameters used to filter out the tokens are the minimum length and maximum length. The parameters define the range for selecting the tokens. In the proposed model the minimum length was set to 4 characters and maximum length to 25 characters i.e. tokens with less than 4 characters and more than 25 characters were discarded.

Stemming

Stemming defines a technique that is used to find the root or stem of a word. The filtered token set undergoes stemming to reduce the length of words until a minimum length is reached. This

resulted in reducing the different grammatical forms of a word to a single term. The basic stemming process can be summarized under two main headings:

- **Removing the endings:**

The general rules for dropping the endings from words include:
i. If a word ends in '*es*' drop the *s*.
ii. If a word ends in '*ing*', delete the *ing*unless the remaining word consists of a single letter or *th*.
iii. If a word ends in a consonant, other than *s*, followed by *s* then delete *s*.

- **Transforming the words:**

The words can be transformed to some other grammatical form using a set of defined rules. For example, if the word ends with '*ies*' but not '*eies*' and '*aies*' then the '*ies*' can be replaced with a '*y*' such as '*Butterflies*' can be replaced with '*butterfly*'.

Figure 4 presents the example words and their stem.

Words	Stem
User, used, using, users	*Use*
Engineering, engineer, engineered	*engineer*
Architectural, architectural, architecturally	*architectur*

Figure4. Different Grammatical forms of a Word and the Corresponding Stem

The stemming technique increases the efficiency and effectiveness of the information retrieval and text mining processes. Matching the similar words results in improved recall rate and also reduces the indexing size as much as 40-50%.

3.3 Features selection

The schemes used for word vector creation includes: Term Occurrence, Binary term occurrence, Term frequency and TF-IDF (term frequency-inverse document frequency).

These are based on the following values:

f_{ij}: total occurrences of the term *i* in the document *j*.
fd_j: total number of terms occurring in document *j*.
ft_i: total number of documents in which the term *i* occurs.

Term occurrence: defines the absolute number of occurrences of a term.

$$Term\ occurence = fij$$

Term frequency:defines the relative frequency of a term in the document.

$$Term\ frequency = fij\ /fdj$$

Binary term occurrence*:* term occurrence is defined as the binary value.

$$Binary\ Term\ Occurrence = 1\ for\ fij>0\ and = 0\ otherwise.$$

TF-IDF*:* it describes how important a word is for a document. It consists of two parts: term frequency (TF) and invert document frequency (IDF).

$$TF\text{-}IDF= (fij\ /fdj)log(1/fti).$$

4. N-GRAMS

An n-gram defines a subsequence of n items from a given sequence. It is used in various fields of natural language processing and genetic sequence analysis. An n-gram model defines a method for finding a set of n-gram words from a given document. The commonly used models include unigrams (n=1), bigrams (n=2) and trigrams (n=3). However the value of n can be extended to higher level grams. The n-gram model can be better explained with the following examples:

Text: “Honesty is the best policy.”
Unigrams: “honesty”, “is”, “the”, “best”, “policy”.
Bigrams: “honesty is”, “is the”, “the best”, “best policy”.
Trigrams: “honesty is the”, “is the best”, “the best policy”.

Unigrams presents the simplest model for the n-gram approach. It consists of all the individual words present in the text. The bigram model defines a pair of adjacent words. Each pair of words forms a single bigram. The higher order grams can be formed in the similar way by taking together the n adjacent words. Higher order n-grams are more efficient in capturing the context as they provide better understanding of the word position.

5. CLASSIFICATION

Machine learning approaches simulate the way humans learn from their past experiences to acquire knowledge and apply it in making future decisions. These learning techniques are widely used in artificial intelligence and document classification. The classification using machine learning can be summed up in two steps:

1. Learning the model using the training dataset
2. Applying the trained model to the test dataset.

Sentiment analysis is a text classification problem and thus any existing supervised classification method can be applied. Our work uses the Naive Bayes classifier and Support Vector Machines for classifying the movie reviews and compares the results obtained using the two approaches.

Naïve Bayes classifieris a simple probabilistic classifier that is based on the Bayes theorem. This classification technique assumes that the presence or absence of any feature in the document is independent of the presence or absence of any other feature. Naïve Bayes classifier considers a document as a bag of words and assumes that the probability of a word in the document is independent of its position in the document and the presence of other word. For a document d and class c:

$$p(c \mid d) = \frac{p(d \mid c)P(c)}{p(d)}$$

Support vector machineshave been the most efficient way for document classification. These are large margin classifiers and perform better than Naïve Bayes and Maximum Entropy in almost all cases. The basic idea behind SVM classification is to find a maximum margin hyperplane that separates the document vector in one class from the other with maximum margin. In this work the Initial tests were carried out using the Naïve Bayes classifier and the Linear Support Vector Machine. Later the linear support vector machine was used to train the model for obtaining the results for n-grams (n=1, 2, 3, 4). Pang, Lee and Vaithyanathan used the similar technique to classify the movie reviews as positive or negative. Our model uses 100000 iterations to obtain the result.

6. RESULTS AND DISCUSSIONS

The dataset used for the experiments was divided into two classes, positive and negative. For a given classifier and a document there are four possible outcomes: true positive, false positive, true negative and false negative. If the document is labelled positive and is classified as positive it is counted as true positive else if it is classified as negative it is counted false negative. Similarly, if a document is labelled negative and is classified as negative it is counted as true negative else if it is classified as positive it is counted as false positive. Based on these outcomes a two by two confusion matrix can be drawn for a given test set.

The confusion matrix in figure 5 forms the basis for the calculation of the following metrics.

i. *Accuracy* = (tp+tn)/ (P+N)
ii. *Precision* = tp/ (tp+fp)
iii. *Recall/ true positive rate* = tp/P
iv. *F-measure* =2/ ((1/precision)+(1/recall))
v. *False alarm rate/ false positive rate* = fn/N
vi. *Specificity* = tn/ (fp+tn) = (1-fp rate)

		p	*True values*	n
Predicted_values	P'	True Positive(tp)		False Positive(fp)
	N'	False Negative(fn)		True Negative(tn)
	Column totals:	***P***		***N***

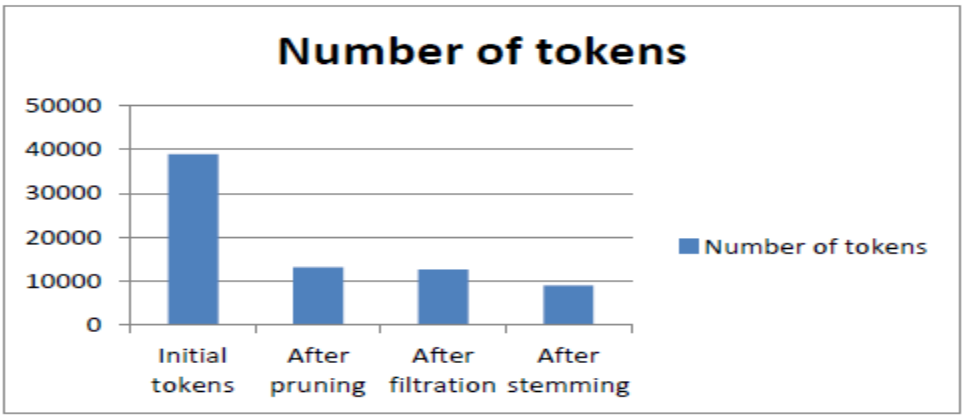
Figure5. Confusion Matrix.

The experiments show that Term frequency-Inverse document frequency (TF-IDF) scheme gives maximum accuracy for linear SVM. Term occurrence gives maximum accuracy for Naïve Bayes classifier. Binary term occurrence also gives the similar results as term occurrence. It is also observed that even on varying multiple parameters linear SVM gives better results than Naïve Bayes. The results obtained using these classifiers for unigrams are summarized in figure7 and figure8.

The dataset consists of 2000 reviews equally divided into 1000 positive and 1000 negative. Initially the wordlist generated for the dataset consist of 38911 tokens. The preprocessingalgorithm explained in figure 3 was applied to the dataset to reduce the number of tokens. The results obtained for the various pre-processing stage are shown in figure6.

Pre-processing	Number of tokens
Initial tokens	*38911*
After pruning	*13197*
After filtration	*12618*
After stemming	*9007*

6 (a). Tabular Representation



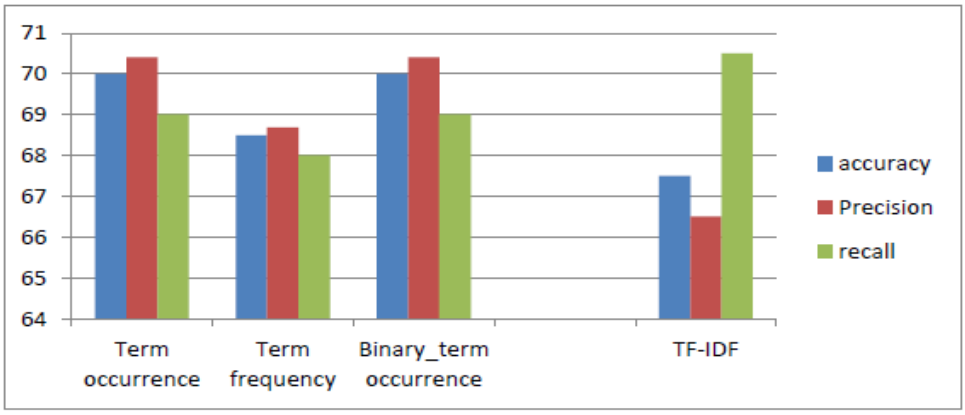
6 (b). Graphical Representation

Figure6. Data Pre-processing Results

The generated tokenset after pre-processing was classified using Naïve Bayes and Linear SVM classifiers. Four different feature selection schemes (TO, TF, BTO, TF-IDF) were used resulting in a total of 8 (2X4) experiments for this step. The results obtained for the two classifiers using these feature selection schemes are shown in figure7 and figure8.

	accuracy	Precision	recall
Term occurrence	70.00	70.41	69.00
Term frequency	68.50	68.69	68.00
Binary term occurrence	70.00	70.41	69.00
TF-IDF	67.50	66.51	70.50

7 (a). Tabular Results for Naïve Bayes Classifier

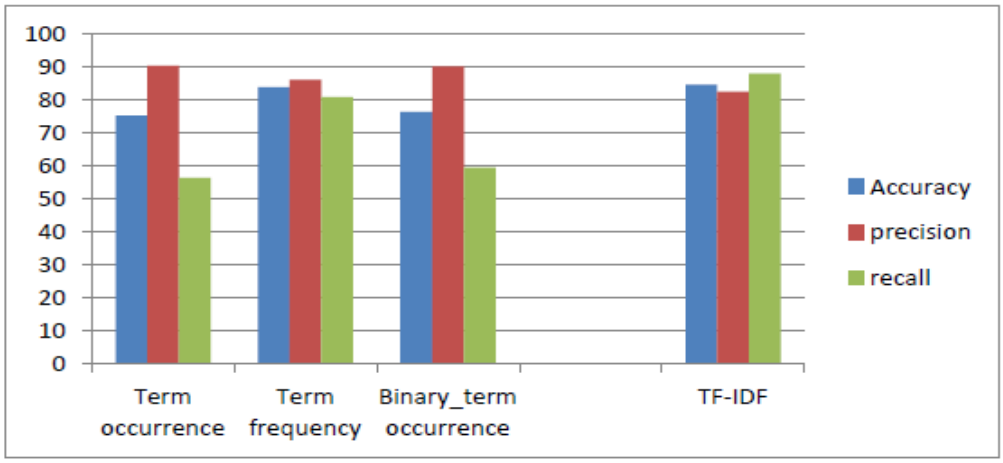


7 (b). Graphical Results for Naïve Bayes Results

Figure7. Results for Naïve Bayes Classifier

	Accuracy	precision	recall
Term occurrence	75.25	90.40	56.50
Term frequency	84.00	86.17	81.00
Binary_term occurrence	76.50	90.15	59.50
TF-IDF	84.75	82.63	88.00

8.(a). Tabular Results for linear SVM classifier



8.(b). Graphical Results for linear SVM classifier

Figure8. Results for Linear SVM Classifier

The results show that linear SVM gives maximum accuracy for TF-IDF scheme therefore the model was trained using the linear SVM and was further used to test a new dataset (large movie review datasetV1.0). The new test set was introduced by Maas et al [22]. The test dataset consists of 10001 positive and 10001 negative movie reviews. The trained model wassuccessful in predicting 9767 negative reviews correctly, giving an accuracy of 97.66 percent. 234 reviews were wrongly predicted as positive. The results obtained for the test set are shown in figure 9 and figure10.

	Labelled	Predicted
positive	10001	10235
negative	10001	9767

Figure9. The true and predicted values for the test set

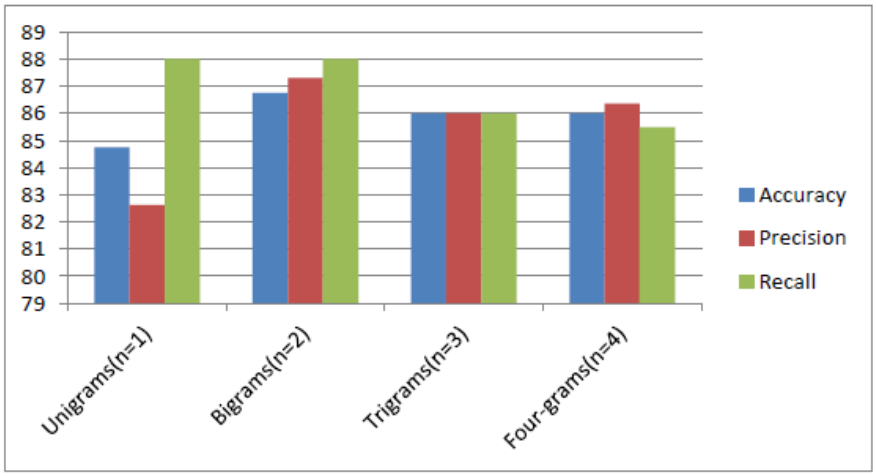
	Minimum	Maximum	Average	deviation
Confidence	0.057	0.898	0.500	0.150
Confidence	0.102	0.943	0.500	0.150

Figure10. The confidence values for the positive and negative predictions

The proposed model trained using Linear SVM and TF-IDF was extended to obtain results for higher order n-grams (n=3, 4). The results obtained for the n-grams are shown in figure11.

	Accuracy	Precision	Recall
Unigrams(n=1)	84.75	82.63	88.00
Bigrams(n=2)	86.75	87.31	88.00
Trigrams(n=3)	86.00	86.00	86.00
Four-grams(n=4)	86.00	86.36	85.50

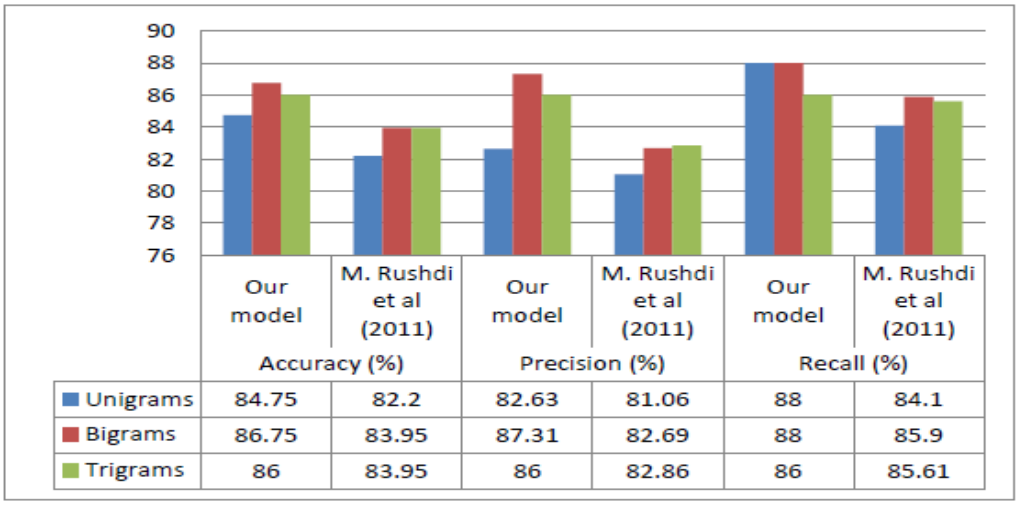
11(a). Tabular results for n-gram



11(b) Graphical Representation for n-gram results

Figure11. Simulation results for n-grams

The maximum accuracy of 84.75% is obtained for unigrams using Linear SVM with TF-IDF scheme. On extending the process for bigrams, trigrams and four-grams we concluded that bigrams gives better accuracy and precision than unigrams while the recall remains the same. The accuracy and precision for bigrams is improved by 2% and 5.32% respectively. The results for trigrams show a fall of .75% in accuracy and a fall of 1.31% in the precision. The recall is also decreased by 2%. The results for 4-grams show a minor improvement in the precision when compared to trigrams while the accuracy remains the same as that for trigrams. The recall is decreased by 0.5%. The generated dataset for trigrams and 4-grams is nearly same due to which the results are nearly the same. Also, the results are also affected by the data domain. Extending the process to higher order n-grams complicate the process leading to over-fitting.



	Our model	M. Rushdi et al (2011)	Our model	M. Rushdi et al (2011)	Our model	M. Rushdi et al (2011)
	Accuracy (%)		Precision (%)		Recall (%)	
Unigrams	84.75	82.2	82.63	81.06	88	84.1
Bigrams	86.75	83.95	87.31	82.69	88	85.9
Trigrams	86	83.95	86	82.86	86	85.61

Figure12. Comparison of Results with M. Rushid et al (2011)

We tested our model using 5-fold cross validation and found out that our proposed model is promising as compared to the previous works carried out using the same dataset and similar techniques. Pang et al [2] obtained the accuracy for unigrams and bigrams using the term presence scheme. They used supervised learning methodologies to classify movie reviews and were successful in achieving an accuracy of 82.90% for unigrams using SVM and feature presence scheme by applying 3-fold cross validations. M. Rushdi et al. [21] obtained the experimental results for SVM and TF-IDF scheme for the same dataset. They also used n-gram approach and performed 3-fold and 10-fold cross validations on the dataset using unigrams, bigrams and trigrams. Our work is closely related to the work done by M.Rushdi et al [21] and shows significant improvements in the results. The comparison of the two works is summarized in figure12.

7. CONCLUSION AND FUTURE WORK

The proposed work presents an approach for sentiment analysis by comparing the different classification methods in combination with various feature selection schemes. We successfully analyzed the different schemes for feature selection and their effect on sentiment analysis. The classification results clearly show that Linear SVM gives more accuracy than Naïve Bayes classifier. Although many other previous works have also identified SVM as a better method for sentiment analysis but our framework differs from previous works in terms of the comparative study of the classification approaches in combination with different feature selection schemes. The results obtained for linear SVM are also better than the previous works. Our results show that the accuracy increases for the bigrams which is in contrast with the results for Pang et al [2]. The affect of varying different parameters is also shown successfully.

The model proposed in this paper is just an initial step towards the improvement in the techniques for sentiment analysis. It is worth exploring the capabilities of the model for the dynamic data and extending the research using hybrid techniques for sentiment analysis. There is considerable scope for improvement in the corpus creation and effective pre-processing and feature selection.

The work can also be extended to improve the results using the naïve bayes classification. Future researches can be carried out to generate better and fast models for higher order n-grams.

REFERENCES

[1] Bing Liu, 2012, Sentiment analysis and opinion mining, Morgan and Claypool publishers.

[2] B. Pang et al, 2002, Thumbs up ?: sentiment classification using machine learning techniques, Proceedings of the ACL-02 conference on Empirical methods in natural language processing, vol.10, 79-86.

[3] P.D. Turney, 2002, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, Proceedings of the Association for Computational Linguistics (ACL), 417–424.

[4] Riloff, E &Wiebe, J., 2003, Learning extraction patterns for subjective expressions, EMNLP'03.

[5] Loren Terveen et al, 1997, PHOAKS: A system for sharing recommendations, Communications of the Association for Computing Machinery (CACM), 40(3):59–62.

[6] Minqing Hu and Bing Liu, 2004, Mining and summarizing customer reviews, Proceedings of the 10th ACM SIGKDD International conference on knowledge discovery and data mining.

[7] Nasukawa, Tetsuya and Jeonghee Yi, 2003, Sentiment analysis: capturing favourability using natural language processing, Proceedings of the K-CAP03, 2nd International Conference on knowledge capture.

[8] Dave et al, 2003, Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews, In Proceedings of the 12th International Conference on World Wide Web, WWW 2003, 519-528.

[9] WiebeJanyce, 1990, Identifying subjective characters in narrative, Proceedings of the International Conference on Computational Linguistics (COLING-1990).

[10] Hearst M., 1992, Direction-based text interpretation as an information access refinement in Text-Based Intelligent Systems, P. Jacobs, Editor 1992, Lawrence Erlbaum Associates, 257-274.

[11] WiebeJanyce, 1994, Tracking point of view in narrative, Computational Linguistics, 233–287.

[12] Hatzivassiloglou et al, 1997, Predicting the semantic orientation of adjectives, Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-1997).

[13] Junichi Tatemura, 2000, Virtual reviewers for collaborative exploration of movie reviews, In Proceedings of Intelligent User Interfaces (IUI), 272–275.

[14] S. Morinaga et al, 2002, Mining product reputations on the web, SIGKDD'02, Edmonton, Alberta, Canada.

[15] P.D. Turney and Michael L Littman,2003, Measuring Praise and criticism: inference of semantic orientation from association, ACM Transactions on Information Systems, TOIS 2003, 21(4), 315-346.

[16] Esuli, A., &Sebastiani, F., 2005, Determining the semantic orientation of terms through gloss classification, In CIKM '05: Proceedings of the 14th ACM international conference on information and knowledge management, 617–624.

[17] Minqing Hu and Bing Liu, 2006, Opinion extraction and summarization on the web, Proceedings of the 21st National conference on Artificial Intelligence, AAAI-2006.

[18] Xiaowen Ding et al, 2008, A holistic lexicon-based approach to opinion mining, WSDM'08, February 11-12, 2008, Palo Alto, California, USA.

[19] Murthy G. and Bing Liu, 2008, Mining opinions in comparative sentences, Proceedings of the 22nd international conference on computational linguistics (Coling 2008), Manchester, August 2008, 241-248.

[20] V. Suresh et al, 2011, A Non-syntactic Approach for Text Sentiment Classification with Stopwords, WWW 2011,March 28–April 1, 2011,Hyderabad, India

[21] M. RushdiSaleh et al, 2011, Experiments with SVM to classify opinions in different domains, Expert Systems with Applications 38.

[22] Andrew L. Maas et al, 2011, Learning Word Vectors for Sentiment Analysis, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 142-150.
[23] Das, Sanjiv and Mike Chen, 2001, Yahoo! for Amazon: Extracting market sentiment from stock message boards, Proceedings of APFA-2001.
[24] Das, Sanjiv and Mike Chen, 2007, Yahoo! for Amazon: Sentiment extraction from small talk on the web, Management Science, 53(9): 1375-1388.
[25] B. Pang and L. Lee, 2008, Opinion mining and sentiment analysis, Foundations and Trends in Information Retrieval 2(1-2), 1–135.
[26] A. Mudinas et al, 2012, Combining lexicon and learning based approaches for concept-level sentiment analysis, Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining, ACM, New York, NY, USA, Article 5, 1-8.
[27] A. Joshi et al, 2011, C-feel-it: a sentiment analyzer for micro blogs, Proceedings of ACL: Systems Demonstrations, HLT '11, 127–132.
[28] ZhongwuZhai et al, 2011, Clustering product features for opinion mining, WSDM'11, February 9–12, 2011, Hong Kong, China.
[29] ZhongwuZhai et al, 2011, Identifying evaluative sentence in online discussions, Association for the advancement of artificial intelligence (www.aaai.org).
[30] Yubo Chen and JinhongXie, 2008, Online Consumer Review: Word-of-Mouth as a New Element of Marketing Communication Mix, Management Science, March 2008, vol 54, no 3, 477-491,.
[31] Wiebe, Janyce et al, 1999, Development and use of a gold-standard data set for subjectivity classifications, Proceedings of the Association for Computational Linguistics (ACL-1999).
[32] WiebeJanyce, 2000, Learning Subjective Adjectives from Corpora, American Association for Artificial Intelli-gence (www.aaai.org).
[33] Won Y. Kim et al, 2009, A method for opinion mining of product reviews using association rules, ICIS 2009, November 24-26, Seoul, Korea.
[34] Hu, Nan et al, 2006, Can online reviews reveal a product's true quality?: empirical findings and analytical modeling of Online word-of-mouth communication, Proceedings of Electronic Commerce (EC).
[35] Kim S. and Hovy E, 2004, Determining the Sentiment of Opinions, COLING'04, Geneva.

Authors

GautamiTripathi received her Bachelor of Engineering (B.Tech) degree in 2012 from GobindBallabh Pant Engineering College (GBPEC), PauriGarhwal, Uttarakhand, India and her Masters degree in 2014 from Galgotias University, Greater Noida, UttarPradesh, India. Her current research interests lies in the area of text mining, data analytics and machine learning**.**

Mr. Naganna S. received the B.E. (CSE) degree in 1999 from Gulbarga University, Gulbarga, Karnataka, India and M.Tech (CSE) degree in 2006 from Visveswaraya Technological University, Belgaum, Karnataka, India. His current research interests include data mining, machine learning, databases, soft computing and distributed computing. He is an IBM certified for DB2 9 Database and application fundamentals and Rational Application Developer (RAD).