GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

SENTIMENTAL ANALYSIS USING TWEETS

A Report for the Final Evaluation of Project

Submitted by

PRANJAL MISHRA

(1613101495)

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

WITH SPECIALIZATION OF DATA ANALYTICS

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of

MR DEEPENDRA RASTOGI, M.Tech., Ph.D.,

Professor

APRIL / MAY- 2020

GALGOTIAS UNIVERSITY

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report "SENTIMENTAL ANALYSIS USING TWEETS" is the bonafide work of "PRANJAL MISHRA(1613101495)" who carried out the project work under my supervision.

SIGNATURE OF HEAD

Dr.MUNISH SABARWAL

PhD (Management), PhD (CS)
Professor & Dean,

SIGNATURE OF SUPERVISOR

Mr. DEEPENDRA RASTOGI, M.Tech., Ph.D.,
Professor
School of Computing Science & Engineering

TABLE OF CONTENTS

LIST OF TABLES (b)

LIST OF ABBREVIATIONS (c)

1.MNB Multinomial Naïve Bayes 2. LDA Latent Dirichlet Algorithm

3. NLP Natural Language Processing 4. SWN Senti Word Net

5. RFA Random Forest Algorithm 6. SVM Support Vector Machines

7. NBC Naïve Bayes Classifiers 8. SNS Social Networking Sites

9. WEKA Waikato Environment for Knowledge Analysis

10. CSV Comma Separated Value file 11. ARFF Attribute Relation File Format

ABSTRACT 1

The 21st century has brought in lots of product in form of movies, software, video games. With the advent of web 2.0, the number of Social Networking Sites (SNS) has increased manifold. With these has increased the volume of users generated content. People make opinion or judgement over lot of above mentioned products and often express it over internet through use of SNS. In daily life people take opinion of their friends and are influenced by them in their decision making process. Opinion is the view or judgement about something. Sentiment Analysis or Opinion Mining is the computational analysis of

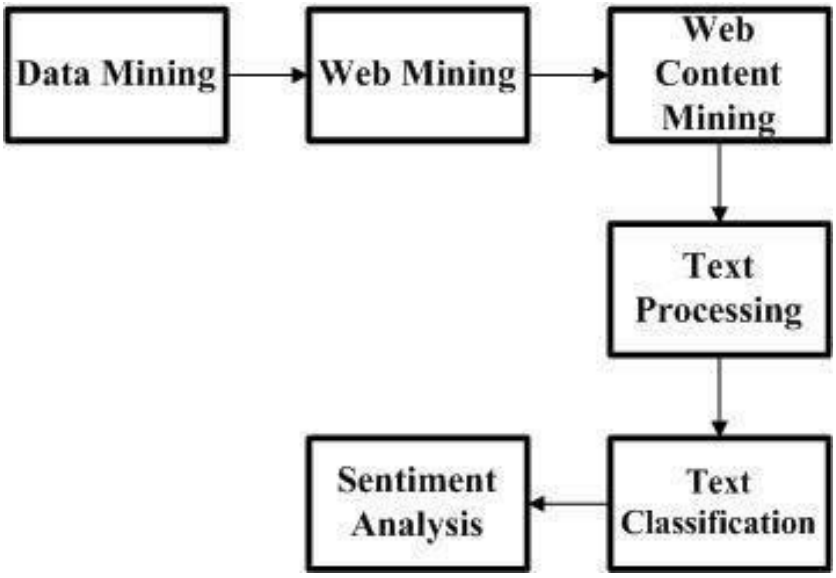
public opinions and sentiments towards a particular subject. Millions of users share their opinions on Twitter, making it a valuable platform for tracking and analysing public sentiment. Such tracking and analysis can provide critical information for decision making in various domains. With the growing availability and popularity of opinion-rich resources such as online review sites and personal blogs, new opportunities and challenges arise as people now can, and do, actively use information technologies to seek out and understand the opinions of others.

In this research, firstly the dataset extracted from the twitter website, is pre-processed. After that feature vector is extracted and feature vector list is selected and lastly classification techniques are applied. These tweets can be classified into positive or negative on the basis of their sentimental orientation. Methods used to classify the tweets are semantic and machine learning approaches. These methods are discussed and applied in this research work. A method for classification which is combination of SentiWordNet and Machine Learning algorithms is also proposed. All of these methods are then analysed on the basis of precision, recall and F-measure. The proposed algorithm gives 86% accuracy as compared to 85% accuracy of Multinomial Naïve Bayes and gives accuracy up to 77% for real time data. The tools used in realization of the proposed work is Java and WEKA primarily. File system is used to store the data as CSV and ARFF files.In order to fulfil the objectives of the proposed research, first literature survey is done to identify the problems then 'Sentiment Analysis' is explained in relevant detail, along with proposed methodology and approaches used and reasons for their selection. Thereafter Design and development is shown followed by validation and conclusion.

INTRODUCTION 5

There has been an exponential growth in the use of online resources, in particular social media and micro-blogging websites such as Twitter over the past decades. These resources offer a rich mine of marketing knowledge to organizations. This project focuses on implementing a classifier using machine learning algorithms to extract sentiment of tweets. A major focus of this study was on comparing different machine learning algorithms based upon their performances. Also, this approach allows

to give a grade to the tweets based upon their intended sentiments which belong to one of the classes namely: negative, neutral, positive. From the evaluation of this study it can be concluded that the proposed machine learning techniques are effective and practical methods for sentiment analysis.



- With the advent of 21st Century, we have witnessed rise of the Social Networking Sites (SNS) like Facebook, Twitter, Google plus etc.
- Now Because of this, there is an overwhelming rise in the user generated contents. People like to share their opinions, thoughts and views.
- The opinions given by the users on products can be used to make important business-related decisions as they give insight into product reception and quality.
- The social media giants like twitter attracts millions of online users sharing their opinions in form of 'tweets'.

PROBLEM IDENTIFICATION

Sentiment Analysis or Opinion Mining can be effectively used in monitoring the SNS, since it allows the user, an insight into the public opinion on specific topics. In our daily life we take opinion of our friends and are influenced by them in our decision making process. Opinion is the view or judgement about something. With the advent of web 2.0, the number of Social Networking Sites (SNS) has increased manifold. With these has increased the volume of users generated content. Sentiment Analysis or Opinion

Mining is the computational analysis of public opinions and sentiments towards a particular subject. Sentiment Analysis is emerging area of Natural Language Processing (NLP) with research which ranges from document level classification to classifying words and phrases by learning their polarity . The research has been done on correlating the relations between public sentiment and real-life events (e.g., polls , stock market . It has been reported that events that happen in real life indeed have a significant effect on the public sentiment over SNS like Twitter. Based on such correlations, some other work , made use of the sentiment signals in blogs and tweets to predict movie sales and elections. The pioneering work of figuring out application and challenges in the field of Sentiment Analysis was presented by Pang and Lee and Liu . They mentioned the techniques used to solve each problem in Sentiment Analysis. In authors have used SentiWordNet as source and applied to the Machine learning classifier. In authors have shown use of machine learning algorithms to classify the documents and comparing results of Naïve Bayes, Maximum Entropy, and SVM with varying n-grams. In another research work , authors have used Machine Learning based classification approaches along with the Unsupervised Semantic Orientation based algorithms for sentiment analysis of movie review texts. In research work of , authors have used Machine Learning based algorithms along with WordNet to classify the tweets. By observing above mentioned literature, it is evident that there is scope of improvement in following areas:

1) Effectiveness of Sentiment Analysis greatly depends on 'Preprocessing of the dataset'. Therefore, text preprocessing should be improved.

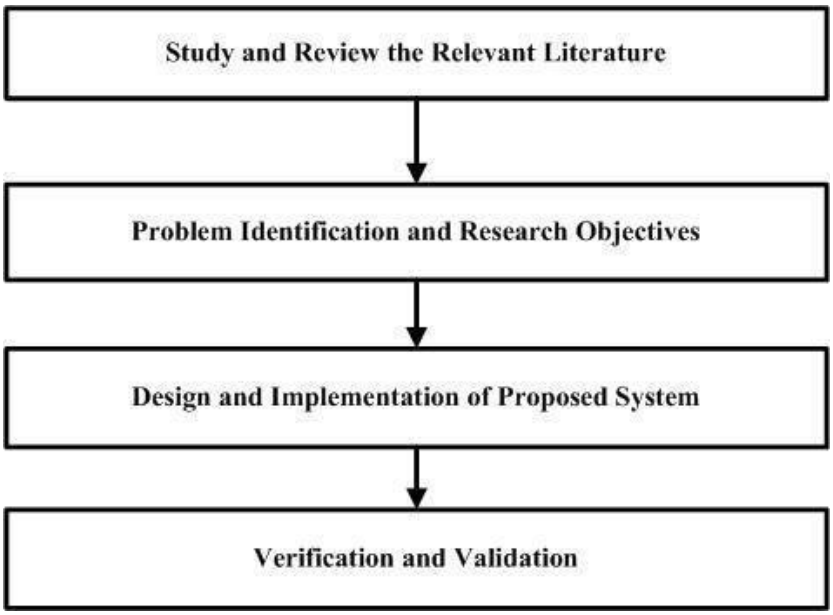
2) Sentiment labelling has been done in various ways, the proposed research should improve the task of sentiment labelling by introducing a new approach.

3) Previous research is mostly focussed on offline data. Real time Sentiment Analysis offers a new challenge for the proposed research.

RESEARCH METHODOLOGY

The approach relies on the combination of two approaches of classification. One being the machine learning approach, which relies on machine learning algorithms to solve sentiment analysis problem as text classification problem. Most common approach to solve this problem is use of supervised learning, where in labelled dataset is used to train

the classifier. The model generated can be used to predict the class of the text. Second approach is the study of the meanings of words and phrases in language. The semantic classification technique implemented by this paper makes use of dictionary based approach where in lexical resource (SentiWordNet) is used. SentiWordNet is a lexical resource in which each synset of WordNet is associated to three numerical scores Obj(s), Pos(s) and Neg(s) ranges from 0.0 to 1.0. One of the tools used in the proposed research are WEKA (Waikato Environment for Knowledge Analysis) which is an open source machine learning library written in Java. The other tool used is Eclipse, which is an open source IDE and platform for building java applications. Eclipse provides a wide variety of plugins and other development-oriented tools. Four main components of research methodology is shown in Figure below shows., first is study and review of the relevant literature, second is problem identification and research objectives, third is design and implementation of the proposed system and last is the verification and validation of the results.



SCOPE OF RESEARCH

The proposed research will help to find how public perception of a certain brand

changes positively or negatively. It will help in Detection of VOC (Voice of Consumer) and VOM (Voice of Market) such information as early as possible helps in direct and target key marketing campaigns. Online commerce sites can use the proposed research to make the policy changes and do Brand Reputation Management (BRM).

OBJECTIVE

Sentiment analysis allows businesses to harness tremendous amounts of free data to understand customer needs and attitude towards their brand. Organizations monitor online conversations to improve products and services and maintain their reputation. The analysis takes customer care to the next level.

It's estimated that 80% of world's data is unstructured and not organized in a predefined manner. Most of this comes from text data, like emails, support tickets, chats, social media, surveys, articles, and documents. These texts are usually difficult, time-consuming and expensive to analyze, understand, and sort through.

Sentiment analysis systems allows companies to make sense of this sea of unstructured text by automating business processes, getting actionable insights, and saving hours of manual data processing, in other words, by making teams more efficient.

- To improve the refinement procedure and pre-processing of the tweets.
- To improve the existing Sentimental labeling and algorithms for sentimental analysis of twitter data-set.
- To give sentiment analysis of Real Time extracted tweets.

SCOPE

The proposed research will help to find how public perception of a certain brand changes positively or negatively. It will help in Detection of VOC (Voice of Consumer) and VOM (Voice of Market) such information as early as possible helps in direct and target key marketing campaigns. Online commerce sites can use the proposed research to make the policy changes and do Brand Reputation Management (BRM).

By observing above mentioned literature, it is evident that there is scope of improvement in following areas:

1) Effectiveness of Sentiment Analysis greatly depends on 'Preprocessing of the dataset'. Therefore, text preprocessing should be improved.

2) Sentiment labelling has been done in various ways, the proposed research should improve the task of sentiment labelling by introducing a new approach.

3) Previous research is mostly focused on offline data. Real time Sentiment Analysis offers a new challenge for the proposed research.

LITERATURE SURVEY 10

Sentiment Analysis is emerging area of Natural Language Processing (NLP) with research which ranges from document level classification to classifying words and

phrases by learning the polarity. The research has been done on correlating the

relations between public sentiment and real-life events (e.g., polls, stock market

[5]. It has been reported that events that happen in real life indeed have a significant

effect on the public sentiment over SNS like Twitter. Based on such correlations, some other work made use of the sentiment signals in blogs and tweets to predict movie sales and elections. The pioneering work of figuring out application and

challenges in the field of Sentiment Analysis was presented by Pang and Lee [8] and

Liu They mentioned the techniques used to solve each problem in Sentiment

Analysis. In authors have used SentiWordNet as source and applied to the Machine learning classifier. In authors have shown use of machine learning algorithms to classify the documents and comparing results of Naïve Bayes, Maximum Entropy, and SVM with varying n-grams. In another research work authors have used ML based classification approaches along with the Unsupervised Semantic Orientation based algorithms for sentiment analysis of movie review texts. In research work of, authors have used ML based algorithms along with WordNet to classify the tweets. In the research work , Lots of effort has been done to analyse information of SNS, such as sentiment trend analysis of SNS users. Author's aim is to analyse the sentimental influence of posts and compare the result on various topics and different social media platforms thereby measuring the sentimental influence of posts.

Literature survey as mentioned is done to look at a literature (publications) in a surface level, or an Ariel view. It includes the survey of place people and publications in the context of Research. The research papers including journals and conferences scholarly articles has been reviewed to give the much needed knowledge used in making this

proposed research. Many research papers were taken into consideration, their main contributions and proposals were noted down. Looking through the problems faced and their solutions by other researchers has made the authors of this proposed research.

SENTIMENT ANALYSIS

Sentiment Analysis or Opinion Mining is the computational analysis of public opinions and sentiments towards a particular subject. The social media giants like twitter attracts millions of online users sharing their opinions in form of ‘tweets’. These tweets can be classified into positive or negative on the basis of their sentimental orientation. In this chapter, Sentiment Analysis process, techniques used, application and challenges are discussed. Online information is becoming progressively dynamic and the emergence of online social media and user-generated content further aggravates this experience. It is hard for a person or an organization to get the latest trends and outline the general opinions about products due to the huge diversity and size of social media, and this builds the need of automated and real time opinion extraction and mining. There are number of articles presented every year in the Sentiment Analysis field. The number of articles in this has increased manifold. This creates a need to have survey papers that summarize the recent research trends and directions of Sentiment Analysis. The data sets which are used is an important part of Sentiment Analysis. The main sources of data are from the product reviews. The reviews given by the users gives insight into product reception and quality, which can be used to make important business related decisions. The reviews sources are mainly e-commerce sites, in which customers review can review a product used or bought by them. With the increased surge in user generated tweets, Twitter has become a hugely popular SNS which allows millions of users to exchange their opinions. Sentiment analysis performed over tweets have been effective and economical in exposing public

sentiments. An organization can visualize users' feedback towards its products by checking the public sentiment in tweets. A political party, by studying public sentiments, can adjust its position corresponding the sentiment changes of the public.

SENTIMENTAL ANALYSIS PROCESS

Step 1: Data Extraction

The data available can be fetched from the E- Commerce websites from their product review page. The data from the SNS can be extracted using Application Programming Interface (API). For example in the case of Twitter we can extract data using Twitter API. This datum is stored in the database for further processing.

Step 2: Pre-processing Pre-processing is the process of cleaning the data readying the text for classification. Online texts contain usually lots of noise and unnecessary parts such as tags, scripts. Pre-processing the data reduces the noise in which helps to improve the performance of the classifier. Pre-processing also speeds up the classification process, thus helping in real time Sentiment Analysis.

Pre-processing including data transformations and filtering can significantly improve the performance. As instance, for a system which gives Sentiment Analysis of twitter feeds for English tweets, we propose following pre-processing strategy:

a) Removing Non English words- Since we are focussing only calculating the Sentiment Analysis of the English tweets, we must get rid of the non-English

tweets. b) Removing Uniform Resource Locators (URLs), hashtags, references, special characters- Cleaning the data of hashtags, references, special characters, will help reduce most of the noise. The term 'RT', which often occur in the twitter feeds should also be replaced by null. c) Slang word translation- For this we take help of the internet slang dictionary and replace the slang words into their meaningful format. d) Removing extra letters from words- Words not present in the lexicon and that

have same letter repeated more than two times are reduced to the word with the repeating letter occurring just once. For example, the exaggerated word "Happyyyyyy" is converted to "Happy".

e) Stemming - Stemming is done using Natural Language Tool Kit (NLTK). Stemmer gives the stem word. For example words such as 'waiting', 'waits', 'waited' are replaced with word 'wait'.

Step 3: Sentiment Identification

The aim is to find out the opinionative words or phrases that best describes the context which we are dealing with. Sentiment Word Identification (SWI) is a basic technique in many Sentiment Analysis applications.

Step 4: Feature Selection

Feature selection is mostly integrated in Machine Learning algorithms like SVM, Neural Networks, k-Nearest Neighbours (KNN), etc. as the very first step. Since the main goal of the feature selection step is decreasing the dimensionality of the feature space. Smaller feature space cuts down the computational cost. As a second objective, feature selection also results in reduction in the over-fitting of the learning scheme to

the training data. During this process, it is also important to find a good trade-off between the computational constraints involved when solving the categorization task and the richness of features.

Step 5: Sentiment Classification

Sentiment Classification techniques can be divided into parts namely, Machine Learning approach and lexicon based approach. Machine Learning methods are based on Machine Learning Algorithms. These algorithms when run over the training dataset creates a model. This model can be used for classification tasks. In the classic work have contributed to SC using Machine Learning Techniques. Machine Learning

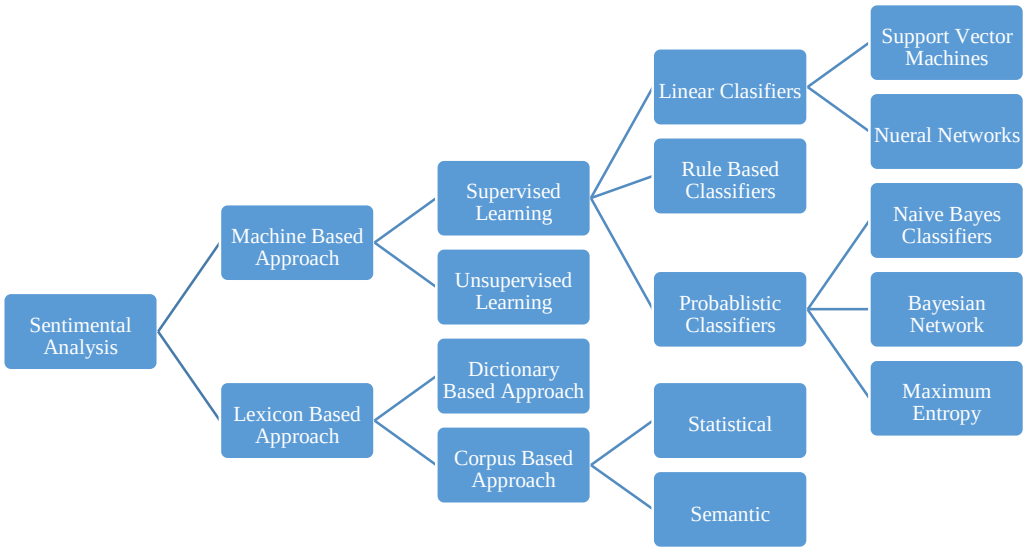
employs in Machine Learning algorithms using linguistic attributes while Lexicon Based approach takes cue from 'sentiment lexicon'. The Machine Learning approach

can be subdivided into two more parts i.e. supervised and unsupervised learning

methods. The supervised learning methods make use of labeled training dataset. In the

case where it is difficult to find the labeled training documents, the unsupervised methods are used. Sentiment extraction involves spotting sentiment words within a particular sentence. This is typically achieved using a dictionary of sentiment terms and their semantic orientations. Dictionary-based approach has some disadvantages associated with them. For example, the sentiment word 'low' in the context of "calories" might have a positive polarity, whereas "low" in the context of "video resolution" is of negative polarity. Taking another example, "go read the book" most likely indicates positive sentiment for book reviews, but negative sentiment for movie reviews.

SENTIMENT CLASSIFICATION TECHNIQUES



Lexicon-Based Approach

The lexicon based methodology ascertains the introduction of a record utilizing the semantic introduction of words or expressions in a specific report. A large portion of the vocabulary based examination utilizes descriptive words as elements for investigating the semantic introduction of content. As indicated by past study, descriptive words are great pointers of Semantic Orientation. For a specific content, the elements are separated and clarified with their Sematic Orientation esteem, by

utilizing the scores given as a part of the word reference. The Semantic Orientation scores are thus aggregated into single sentiment score. Be that as it may, in spite of the fact that a disengaged descriptor may demonstrate subjectivity, there might be lacking connection to decide semantic introduction. As pointed out by author in [20], the descriptor “brutal”, "insane" may have negative introduction in a pet audit, in an expression, for example, "brutal and insane breed of dog", however it could have to a great extent a positive orientation in a film survey, in an expression like "brutal and insane action sequence".

A) Dictionary-based approach

Dictionary-Based approach involves using a dictionary that contains synonyms and antonyms of a word. A small set of opinion words collected mostly manually are then grown by looking up the reference in the well- known thesaurus i.e the dictionary containing the synonyms and antonyms of the words. The newly found words are iteratively added into the set of opiniated words. The process is iteratively repeated until no new word is found.

B) Corpus Based Approach

The Corpus-based depends on syntactic patterns. These patterns occur together along with a seed list of opinion words to find other opinion words in a large corpus. There are two methods in the corpus based approach: Statistical Approach If the word appears intermittently amid positive texts, then its polarity is positive. If it appears frequently among negative texts, then its polarity can be considered as negative. If it has equal frequencies, then it can be considered as neutral word. Seed opinion words can be found using statistical techniques. Most state of the art methods are based on the observation that similar opinion words mostly appear together in a corpus. Thus, if two words appear together frequently within the same context, then there is high probability that they have same polarity. Therefore, the polarity of an unknown word can be determined by calculating the relative frequency of co-occurrence with another word. This could be done using Pointwise Mutual Information (PMI) as in example suggested by [11], SO of a given phrase is calculated by comparing its similarity to a positive word (“Awesome”) And its similarity with negative word (“Awful”). More

explicitly, a phrase is given a numerical rating by taking the mutual information between the given phrase and the positive reference word “Awesome” and subtracting the mutual information between the given phrase and the negative

reference word “Awful”. Using part-of-speech (POS) patterns, this technique then classifies the text by extracting the bigrams. PMI is then calculated by using the polarity score for each

bigram.

 Semantic approach

This rule appoints comparative sentiment values to semantically close words. These Semantically close words can be acquired by getting the list of assumption words and iteratively growing the underlying set with equivalent words and antonyms and afterward deciding the assessment extremity for an obscure word by the relative tally of positive and negative equivalent words of this word [14]. One example of semantic approach is using SentiWordNet. The total score is calculated using Word Sense Disambiguation (WSD) and SentiWordNet. As total score remains in between -1 and 1, where 1 being highly positive and -1 being highly negative. This score can be used

to our advantage to classify tweets into multiple classes to help understand deeply the sentiment value.

Machine Learning Approach

This approach relies on machine learning algorithms to solve sentiment analysis problem as text classification problem. Most common approach to solve this problem is use of supervised learning, where in labelled dataset is used to train the classifier. The model generated can be used to predict the class of the text. The machine learning approach is explained in subsequent stages.

A) Naive Bayes Classifier

Naïve Bayes is the popular method for the text classification. Introduced in 1960s, Naïve Bayes is still very well suited for document classification to detect spam,

categorizing the text to topics (eg: comedy from tragedy). Although being simple and easy, Naive Bayes classifier can even outperform several sophisticated classification methods in practice. Naïve Bayes Classifier is a probabilistic classifier, which rather than giving output as a likely class, gives the degree of certainty in form of probability distribution. The classifier is based on the bayes theorem and is under assumption that

given the context of the class, all features are independent of each other. This is known as naive Bayes assumption as represented in equation i.e.

$$p\,\vec{F}\,|Ci \quad = \prod_j p\,Fj|Ci \qquad \textbf{(3.1)}$$

Naïve Bayes theorem, represented in equation (3.2)

$$| \quad = \frac{\qquad\qquad}{|} \qquad \textbf{(3.2)}$$

Which can also be written as:

$$\times \qquad = \frac{h}{\qquad\qquad} \qquad \textbf{(3.3)}$$

Where x is represented by a vector = , ... representing some *n* features. Naïve Bayes classifier can be represented as in equation (3.4):

$$\hat{} = \max_{\in\{,,,\}} \qquad \prod^{n}_{=} \qquad | \qquad \textbf{(3.4)}$$

B) Multinomial Naive Bayes

Multinomial Naïve Bayes is highly studied classifier. It is relatively effective, fast and easy to implement. The earliest description of this classifier can be found in [14]. Multinomial Naive Bayes is designed in a way that it works better with the text classification. The multinomial model captures word frequency information in

documents. The [15] does a very coherent work in explaining the application to text classification. For a Feature Vector = , , ... where is the probability that event i occurs and is the number of times event I was observed. The multinomial naive Bayes classifier can be expressed as in equation (3.5):

$$\log \quad | \quad \propto \log \quad \prod_{=} \quad = \log \quad + \sum_{=} \quad . \log \qquad \textbf{(3.5)}$$

C) Support Vector Machine

A Support Vector Machine (SVM) is formally defined by a separating hyperplane. It is a discriminative classifier. Although SVM features a linear decision boundary, it is very resilient to overfitting. Figure 3.4 shows the two classes separated via hyperplane along with equation, one class is being represented by disc shaped objects and other being represented by triangular objects.

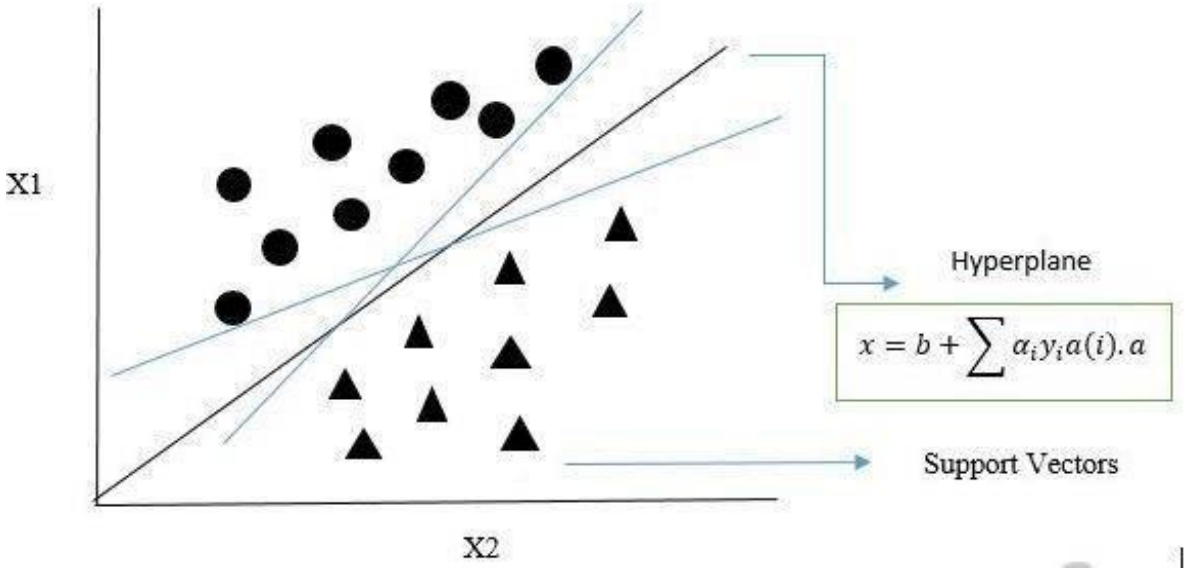


Figure 3.4: Using SVM to separate two different classes via hyperplane

D) Random Forest

Random Decision Forest technique was first proposed in research work of [16] as the ensemble learning method for classification, regression which works by creating a multitude of decision tress. The algorithm for inducing random forest was developed by Leo Breiman in research work [17]. According to him "The Random Forest classifier is a collection of tree classifiers {h(x,Θk), k=1, ...} where {Θk} represents distributed random vectors which are independent and identical.

3.4 CHALLENGES IN SENTIMENTAL ANALYSIS

The main challenges that are faced by OM and Sentiment Analysis are the following:

A) Detection of spam and fake reviews

The Sentiment Analysis process is jeopardized when it encounters the fake data. Since web contains spam and fake data as well, these unnecessary elements should be taken care of for effective sentiment classification. **B) Domain- independence:**

One more challenge that Sentiment Analysis encounters is the Domain problem. One research may work well with a particular domain but it does not prove to be effective in the other domain.

C) Sarcastic sentences

Content may have Sarcastic and humorous sentences. For instance, "Motion picture was awesome to the point that I needed to rest through it to overlook it." In such case, positive words can have negative feeling of importance. Mocking or humorous sentences can be difficult to distinguish which can prompt mistaken assessment mining. Research has been concentrated on the part that distinctive lexical elements play, for example, interpositions (e.g."gee" or "gosh") and accentuation images (e.g., '?') in perceiving mockery in stories. Lukin et al. in [26] investigated the capability of a bootstrapping technique for mockery order in social exchange to learn lexical N-gram signals connected with mockery (e.g., "goodness truly", "I get it", "no chance", and so forth.) and in addition lexico-syntactic examples.

D) Knowledge Base

Sentiment analysis in order to work effectively should also be aware of world's facts, figures and news. This can be achieved by maintaining a proper database with all knowledge related information. However it is possible to maintain knowledge base for a particular domain, it is very hard to do it for multiple domains.

3.5 APPLICATIONS OF SENTIMENTAL ANALYSIS

Sentiment Analysis can be used in diverse fields for various purposes. This section discusses some of the Common ones.

A) Online Commerce

Sentiment Analysis has found its most important application in online commerce. Users submit their experience about shopping on the E- Commerce websites by reviewing their thoughts, opinions and their take on product qualities. Sometimes the users put a rating with numerical scores. Sentiment Analysis can pull this information for novel use. Companies on the other hand can gain from this information by improving upon their products or services.

B) Voice of Customers (VOC) and Voice of Market (VOM)

VOC is a market research technique to describe the in-depth process of capturing a customer's intentions, desires, antipathies and expectations while VOM means that you would be surveying not only your own customers but those of key competitors as well.

C) Brand Reputation Management(BRM)

BRM helps in finding how public perception of a certain brand changes positively or negatively. The variation after an event can be analysed using Sentiment Analysis. A novel work in interpreting the change in sentiment variation has been done by Tan S et al. [24]

D) Recommendation Systems

The Sentiment Analysis can be used to create Recommendation System [5]. This system recommends after going through the public opinions, which one should be recommended and which ones not.

E) Policy Making

Using Sentiment Analysis, policy makers can go through the public sentiment towards a policy, and use it to make the policies which are in demand by the public at large [28].

SUMMARY

In this chapter, term ‘Sentiment Analysis’ is explained. The process of Sentiment Analysis and classification techniques used to achieve this are also explained. Challenges in Sentiment Analysis along with applications are also briefly discussed.

API that are used are WEKA, Twitter4J and Stanford POS tagger. WEKA is a popular suite for Machine Learning software, Twitter4j is a library toolkit to integrate java application to the twitter service, Stanford POS tagger is a piece of software that reads text in some language and assigns parts of speech to each word.

Features in our system are extracted using unigram in the case of Machine Learning and using POS_tags in the case of Semantic Analysis. CLASSIFIER used is multinomial Naive Bayes. LEXICAL RESOURCE used is sent wordnet.

Working:

User fires a query through search box. The system makes use of 'Twitter4J' API along with proper credentials to login into the Twitter. Number of tweets that user wants can be set. The extracted tweets are then stored in Comma Separated Value (CSV) file.

Feature are extracted using unigram model in case of Machine Learning Approach and implicit tokenizer of Stanford POS tagger. Then WEKA API is used to train classifier using training dataset. This paper uses the labelled dataset available online on the link given at the end-note .This creates a classifier model. This is then sent to get sentiment scores using POS tagging and Sent WordNet algorithm. The user can get the results

MACHINE LEARNING APPROACH:

This approach relies on machine learning algorithms to solve sentiment analysis problem as text classification problem. Most common approach to solve this problem is use of supervised learning, where in labelled dataset is used to train the classifier. The model

generated can be used to predict the class of the text. The machine learning approach is explained in subsequent stages.

4.2.1 Classifier Selection

This paper takes into consideration four types of classifiers namely Naïve Bayes Classifier (NB), Multinomial Naïve Bayes Classifier (NBM), Random Forest (RF) and Support Vector Machine (SVM). These classifiers are on the training dataset and evaluated on the basis of Precision, Recall and F-measure. The results in table 4.1 and graph 4.1 are over the 10- Fold Cross Validation process on labelled dataset of 50,000 tweets.

Table 4.1: Result of 10- Fold Cross Validation process on labelled dataset

	Precision	**Recall**	**F-Measure**
NB	0.771	0.783	0.777
NBM	0.834	0.851	0.837
RF	0.811	0.833	0.778
SVM	0.819	0.841	0.822

```
Correctly Classified Instances       39091               78.2524 %
Incorrectly Classified Instances     10864               21.7476 %
Kappa statistic                          0.1931
Mean absolute error                      0.2723
Root mean squared error                  0.399
Relative absolute error                 95.9457 %
Root relative squared error            105.9306 %
Coverage of cases (0.95 level)          97.6939 %
Mean rel. region size (0.95 level)      87.3086 %
Total Number of Instances            49955

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.882     0.698     0.859      0.882     0.870      0.194     0.696     0.910     pos
               0.302     0.118     0.345      0.302     0.322      0.194     0.696     0.308     neg
Weighted Avg.  0.783     0.599     0.771      0.783     0.777      0.194     0.696     0.807

=== Confusion Matrix ===

     a     b   <-- classified as
 36512  4891 |     a = pos
  5973  2579 |     b = neg
```

Figure 4.1: 10- Fold Cross Validation on Naïve Bayes Classifier

Figure 4.1 shows 10-fold cross validation on Naïve Bayes classifier. The results show value of precision to be 0.771, recall to 0.783 and f-measure as 0.777. The figure is the snapshot achieved by using the WEKA API.

```
Correctly Classified Instances       42494               85.0646 %
Incorrectly Classified Instances      7461               14.9354 %
Kappa statistic                          0.3886
Mean absolute error                      0.2108
Root mean squared error                  0.3352
Relative absolute error                 74.283  %
Root relative squared error             88.9793 %
Coverage of cases (0.95 level)          98.5247 %
Mean rel. region size (0.95 level)      81.7175 %
Total Number of Instances            49955

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.946    0.612    0.882      0.946    0.913      0.401    0.815     0.949     pos
               0.388    0.054    0.598      0.388    0.471      0.401    0.815     0.521     neg
Weighted Avg.  0.851    0.516    0.834      0.851    0.837      0.401    0.815     0.876

=== Confusion Matrix ===

     a     b   <-- classified as
 39173  2230 |     a = pos
  5231  3321 |     b = neg
```

Figure 4.2: 10- Fold Cross Validation on Naïve Bayes Multinomial Classifier

Figure 4.2 shows 10-fold cross validation on Naïve Bayes Multinomial classifier. The results show value of precision to be 0.834, recall to 0.851 and f-measure as 0.837. The figure is the snapshot achieved by using the WEKA API.

```
Correctly Classified Instances        8323               83.28   %
Incorrectly Classified Instances      1671               16.72   %
Kappa statistic                          0.1389
Mean absolute error                      0.2481
Root mean squared error                  0.35
Relative absolute error                 85.0311 %
Root relative squared error             91.6376 %
Coverage of cases (0.95 level)          99.4697 %
Mean rel. region size (0.95 level)      91.9051 %
Total Number of Instances            9994

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.990    0.898    0.837      0.990    0.907      0.221    0.776     0.933     pos
               0.102    0.010    0.693      0.102    0.178      0.221    0.776     0.454     neg
Weighted Avg.  0.833    0.740    0.811      0.833    0.778      0.221    0.776     0.848

=== Confusion Matrix ===

    a    b   <-- classified as
 8142   80 |    a = pos
 1591  181 |    b = neg
```

Figure 4.3: 10- Fold Cross Validation on Random Forest Classifier

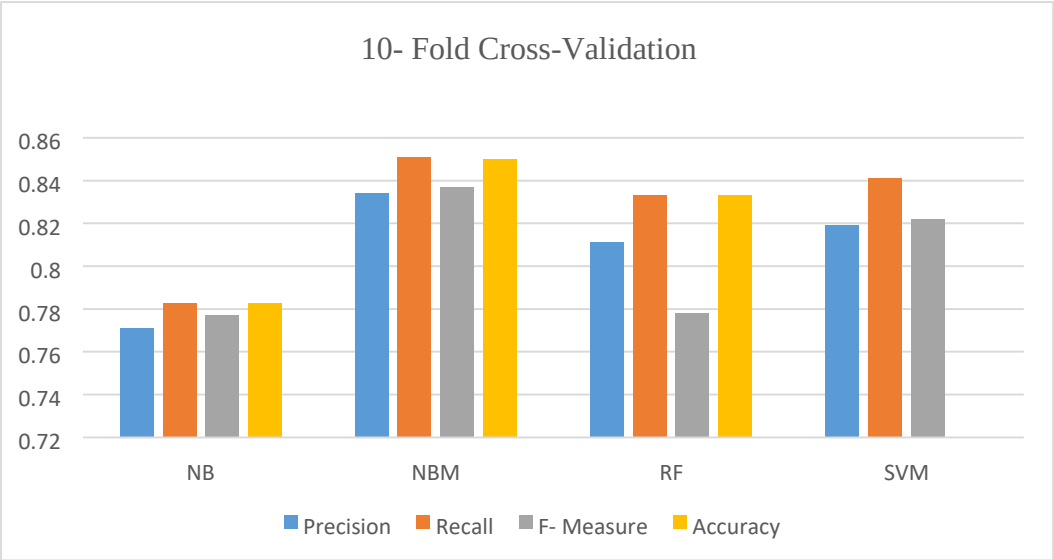
Figure 4.3 shows 10-fold cross validation on Random Forest classifier. The results show value of precision to be 0.811, recall to 0.833 and f-measure as 0.778. The figure is the snapshot achieved by using the WEKA API.

```
Correctly Classified Instances        8405               84.1005 %
Incorrectly Classified Instances      1589               15.8995 %
Kappa statistic                          0.3414
Mean absolute error                      0.159
Root mean squared error                  0.3987
Relative absolute error                 54.4908 %
Root relative squared error            104.4025 %
Coverage of cases (0.95 level)          84.1005 %
Mean rel. region size (0.95 level)      50      %
Total Number of Instances             9994

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.951    0.668    0.868      0.951    0.908      0.361    0.641     0.866     pos
               0.332    0.049    0.592      0.332    0.425      0.361    0.641     0.315     neg
Weighted Avg.  0.841    0.558    0.819      0.841    0.822      0.361    0.641     0.768

=== Confusion Matrix ===

    a    b   <-- classified as
 7817  405 |    a = pos
 1184  588 |    b = neg
```

Figure 4.4: 10- Fold Cross Validation on Support Vector Machine Classifier

Figure 4.4 shows 10-fold cross validation on Support Vector Machine classifier. The results show value of precision to be 0.819, recall to 0.841 and f-measure as 0.822. The figure is the snapshot achieved by using the WEKA API.

Graph 4.1: Coherent view of how the above stated classifiers perform on the stated parameters.



As it turns out that Multinomial Naïve Bayes classifier most suits the requirements in terms of Precision, Recall, F-Measure and Accuracy. The other sophisticated algorithms like Random Forest and SVM do somewhat match up in terms of accuracy but Multinomial Naïve Bayes classifier is much more faster than others which will be suited for real-time system, where test-set is fetched in real time.

4.2.2 Tokenizer Selection

Tokenizer is the way to split the data. The unit of measurement of splitted data is ‘token’. Finding potentially predictive n-grams such as unigrams, bigrams, and trigrams is an important task especially in the Sentiment Analysis process. The different n-grams gives different results according to the situation. When the data such as ‘not heroic’ is encountered, the advantages of seeing the two words together is understandable. The figures 4.5, 4.6 shows results of varying models on multinomial naïve Bayes classifier with 10-fold cross- validation via snapshot using WEKA API.

```
Correctly Classified Instances       41694               83.4631 %
Incorrectly Classified Instances      8261               16.5369 %
Kappa statistic                          0.3764
Mean absolute error                      0.2153
Root mean squared error                  0.3545
Relative absolute error                 75.8818 %
Root relative squared error             94.1074 %
Coverage of cases (0.95 level)          97.2175 %
Mean rel. region size (0.95 level)      78.7869 %
Total Number of Instances           49955

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.917    0.566    0.887      0.917    0.902      0.379    0.800     0.944     pos
               0.434    0.083    0.520      0.434    0.474      0.379    0.800     0.484     neg
Weighted Avg.  0.835    0.483    0.824      0.835    0.829      0.379    0.800     0.865
```

Figure 4.5: Bigram model of 10- Fold Cross Validation on Naïve Bayes Multinomial Classifier

Figure 4.5 shows bigram model of 10-fold cross validation on Naïve Bayes Multinomial classifier. The results show value of precision to be 0.824, recall to 0.835 and f-measure as 0.829. The figure is the snapshot achieved by using the WEKA API.

```
Correctly Classified Instances       41899               83.8735 %
Incorrectly Classified Instances      8056               16.1265 %
Kappa statistic                          0.1812
Mean absolute error                      0.2577
Root mean squared error                  0.3599
Relative absolute error                 90.805  %
Root relative squared error             95.5502 %
Coverage of cases (0.95 level)          99.3534 %
Mean rel. region size (0.95 level)      94.1357 %
Total Number of Instances           49955

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.982    0.857    0.847      0.982    0.910      0.244    0.658     0.893     pos
               0.143    0.018    0.627      0.143    0.233      0.244    0.658     0.347     neg
Weighted Avg.  0.839    0.713    0.810      0.839    0.794      0.244    0.658     0.799

=== Confusion Matrix ===

     a     b   <-- classified as
 40673   730 |     a = pos
  7326  1226 |     b = neg
```

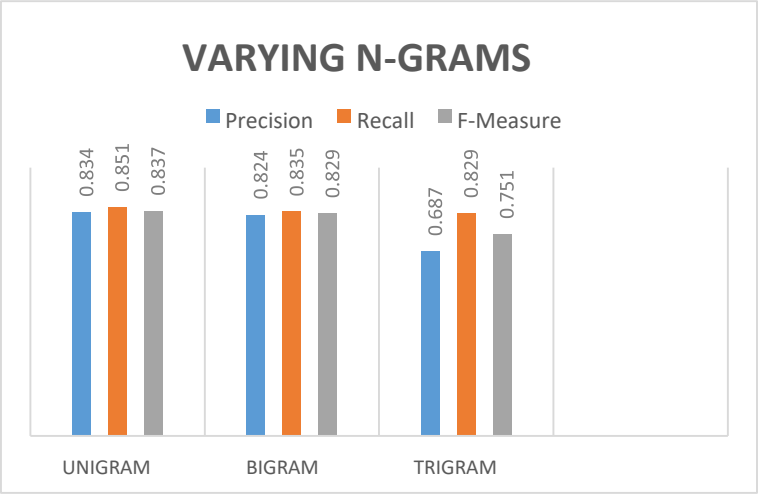
Figure 4.6: Trigram model of 10- Fold Cross Validation on Naïve Bayes Multinomial Classifier

Figure 4.6 shows trigram model of 10-fold cross validation on Naïve Bayes Multinomial classifier. The results show value of precision to be 0.810, recall to 0.839 and f-measure as 0.794. The figure is the snapshot achieved by using the WEKA API. Table 4.2 and graph 4.2 aims to let user visualize how NBM classifier works with unigram, bigram and trigram model

Table 4.2: Results of varying n-grams

	Precision	**Recall**	**F-Measure**
Unigram	0.834	0.851	0.837
Bigram	0.824	0.835	0.829
Trigram	0.687	0.829	0.751

Graph 4.2: Effect of varying n-grams on precision, recall and F-Measure.



From the graph 4.2, it is evident that value of precision decreases from 0.834 to 0.824 and further to 0.687 and value of recall decreases from 0.851 to 0.835 and further to 0.829. While value of F-measure decreases from 0.837 to 0.829 to 0.751. Therefore, it can be concluded that unigram model is best suited for the proposed research. So, the unigram is selected as tokenizer.

SEMANTIC APPROACH

Semantics is the study of the meanings of words and phrases in language. The semantic classification technique implemented by this paper makes use of dictionary based

approach where in lexical resource (SentiWordNet) is used. SentiWordNet is a lexical resource in which each synset of WordNet is associated to three numerical scores Obj(s), Pos(s) and Neg(s) ranges from 0.0 to 1.0.

4.3.1 Tokenizer Selection

Selection of tokenizer in semantic approach plays an important part. Unigram takes one word at a time and lets SentiWordNet to assign sentiment score to it. But in this case, as pointed out in [13] even better approach, where in implicit tokenizer of Stanford Parts Of Speech (POS) tagger is used as for SentiWordNet. The POS tagger reads the text and assigns POS to each word, such as noun, adjective, etc. For the default character encoding of the tagger is UTF-8 (Unicode), so UTF-8 encoding is maintained in the pre-processing. For getting the best results, implicit tokenizer of Stanford POS tags is used.

4.3.2 Classifying using SentiWordNet

The total score is calculated using Word Sense Disambiguation (WSD) and SentiWordNet. As total score remains in between -1 and 1, where 1 being highly positive and -1 being highly negative. This score can be used to our advantage to classify tweets into multiple classes to help understand deeply the sentiment value. Scores used along with their subsequent classes are shown in table 4.3. These scores are classified into various slabs such as Strong_Positive, Strong_negative, and Weak_positive etc.

Table 4.3: SentiWordNet Scores with classification

Sent Score	**Classification**
Above 0.75	Strong_Positive
Between 0.5 and 0.75	Well_Positive
Between 0.25 and 0.5	Positive
Between 0 and 0.25	Weak_Positive
Between 0 and -0.25	Weak_Negative
Between -0.25 and -0.5	Negative
Between -0.5 and -0.75	Well_Negative
Below -0.75	Strong_Negative

SUMMARY

This chapter discusses in detail what approach has been used in this proposed research and reason of doing so. The two approaches machine based approach and semantic approach are discussed in relevant detail. Along with this is discussed what algorithms and parameters are to be selected in order to fulfil the objectives of the proposed research.

DESIGN AND DEVELOPMENT OF PROPOSED SYSTEM

INTRODUCTION

This chapter is divided into three subsections. First process flow of the proposed research is discussed in detail. Then the algorithm used in the process is explained, after which the implementation part is discussed. For designing and developing the application which implements the proposed research, First the process flow of the application is visualised, which gave an insight into how development of the application will take place. Modules to be developed were first visualized and then were implemented. In this chapter, how the process flow of the proposed research works along with the pseudo code of algorithm used and implementation is discussed.

PROCESS FLOW OF THE PROPOSED RESEARCH

The process flow of the proposed system can be visualized by figure 5.1.

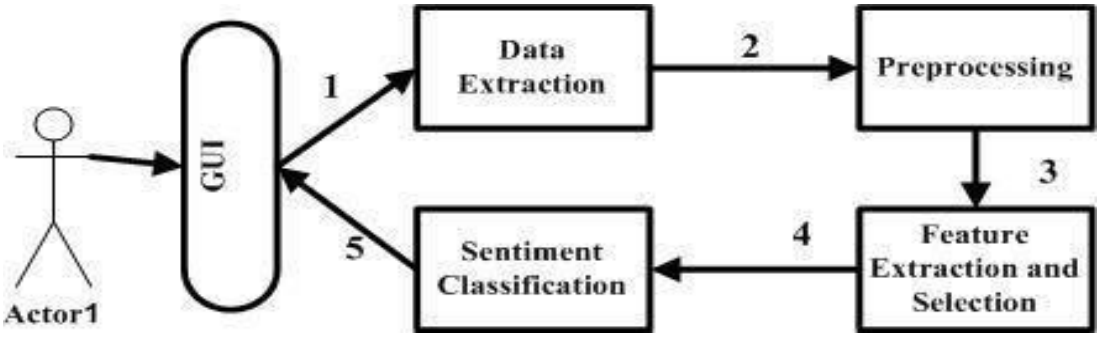


Figure 5.1: The process flow of the proposed system

The steps followed in the realization of the proposed system are explained in following steps:

Step-1: Data Extraction Module

User fires a query through search box. The system makes use of 'Twitter4J' API along with proper credentials to login into the Twitter. Number of tweets that user wants can be set. The extracted tweets are then stored in Comma Separated Value (CSV) file.

Step-2: Preprocessing Module

Pre-processing is the process of cleaning the data and readying the text for classification. The tweets extracted from the twitter API are leeched with irrelevant details, which will do no good in the text classification task. Pre-processing also speeds up the classification process, thus helping in real time SA. In [14], authors have shown that appropriate text pre-processing including data transformations and filtering can significantly improve the performance. The authors propose to eliminate the Uniform Resource Locators (URLs), hashtags, references, special characters and special Twitter Symbols like @, RT etc.

Step-3: Feature Extraction and Selection Module

Feature extraction is process of reducing the amount of resources required to describe a large dataset while the process of selecting the subset of relevant features is known as Feature Selection. Feature extraction and Feature Selection is part the of dimensionality reduction. Features in our proposed system are extracted using unigrams in the case of Machine Learning and using POS tags in the case of Semantic Analysis. The 'words' are selected in case of Machine Learning and 'POS tags' are selected in case of Semantic Analysis. Features are not reduced as it will affect the accuracy of the research.

Step-4: Sentiment Classification

Sentiment Classification is first done using Multinomial Naïve Bayes Algorithm and SentiWordNet Algorithm as discussed in above sections. The proposed algorithm is then used to classify the tweets. The proposed algorithm can be explained as below:

PSEUDO CODE FOR PROPOSED SYSTEM:

Input: Labelled Training-dataset and search query

Output: Sentiment polarity

Step1: Extract the tweets from Twitter API: *// Extracted tweets becomes test set*

Test set 'ts'

Step2- Preprocessing

For tweet t: *// Preprocessing module*

Preprocessing (String t)

Removing URLs, special symbols, Non-English Words

Return t *// Return the processed tweet*

Step3- Train a classifier 'C' on training dataset 'TS' *// Using MNB Classifier*

Return Classifier model

Step4- Extract Feature Vector list and features from *//Feature Extraction and Selection*

For tweet t:

StringToWordVector (t) *// WEKA filter used*

Return Feature Vector

For n in Feature Vector:

Return Features: Words

Step5- Apply Classifier model and get class (pred) of each tweet t. *// pred is String*

Step6- Extract Feature Vector list and features *//Feature Extraction and Selection*

For tweet t:

POS tagger (t) *// Stanford POS Tagger Used*

Return features: tags

Step 7- Get scores using SentiWordNet

Classify into classes (sent) *// sent is a String*

Step 8- For tweet t:

If sent= strong_positive or sent=strong_negative

Return sent; *// Use results from SentiWordNet*

Else if sent = well_positive and pred= pos

Return sent *// Use results from SentiWordNet*

Else if sent= well_negative and pred= neg

Return sent *// Use results from SentiWordNet*

Else Return pred; *// Use results from MNB classifier*

IMPLEMENTATION & WORKING OF SYSTEM 33

For realisation of the proposed approach, an application is designed which makes use of the java code for implementation. Data is maintained in file system using Comma Separated Value (CSV) files. API that are used are WEKA, Twitter4J and Stanford POS tagger. WEKA is a popular suite for Machine Learning software, Twitter4j is a library toolkit to integrate java application to the twitter service. Stanford POS tagger is

explained above. Main modules that have been designed and implemented are explained as below:

a) Data Extraction Module

Figure 5.2 shows the Main window of the application, which asks user to input a 'topic' in search textbox. The three buttons used in the GUI initialises the Machine Learning, SentiWordNet and Proposed algorithm. The application uses 'Twitter4J' API to extract the tweets. These tweets are saved in CSV format in the file system.

Figure 5.2: Application window

b) Pre-processing Module

Saved Tweets in CSV file is pre-processed according to methods discussed above. The CSV file is converted to Attribute Relation File Format (ARFF) to match the training dataset. This is now the 'test set'. Figure 5.3 shows the pre-processed tweets saved in ARFF file.

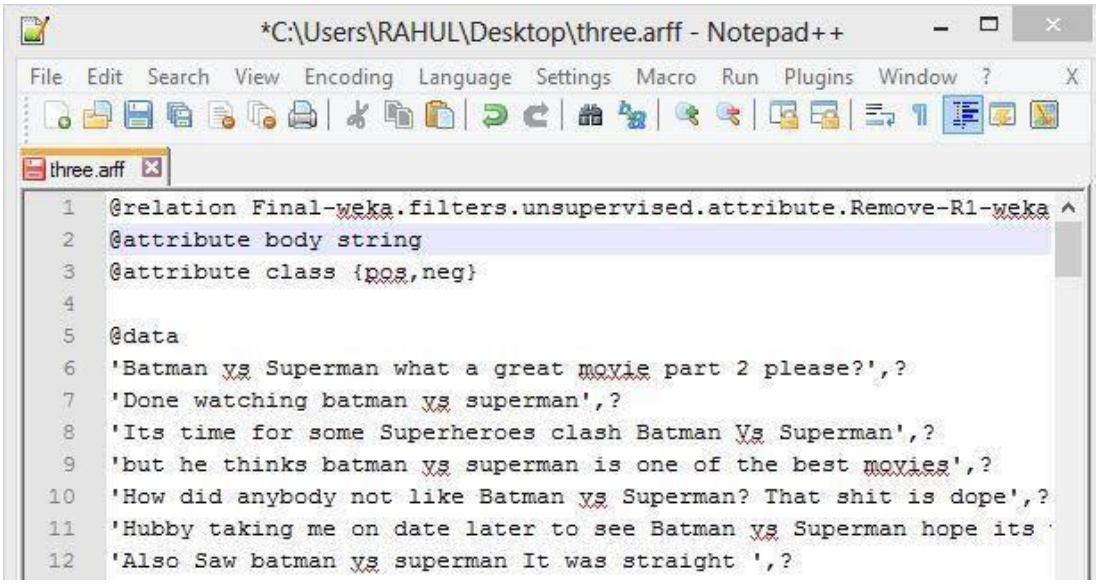


Figure 5.3: Pre-processed data in ARFF file

c) Feature Extraction and Selection Module

Feature are extracted using unigram model in case of Machine Learning Approach and implicit tokenizer of Stanford POS tagger. Then WEKA API is used to train classifier using training dataset. This paper uses the labelled dataset available online on link [15]. Here n-grams and possible features are chosen. This creates a classifier model.

d) Sentiment Classification

The test set is then fed to the model to find out the class. Using same application, if user wants to get results using Semantic approach. The same CSV file is then used to get sentiment scores using POS tagging and SentiWordNet algorithm. The user can then find the sentiments of the same topic using combined approach as discussed above and get the results. The results of the algorithms when run independently can be seen in figure 5.4 and 5.5 and 5.6. In these figures it is clearly seen that tweets are followed by class given by the corresponding approach.

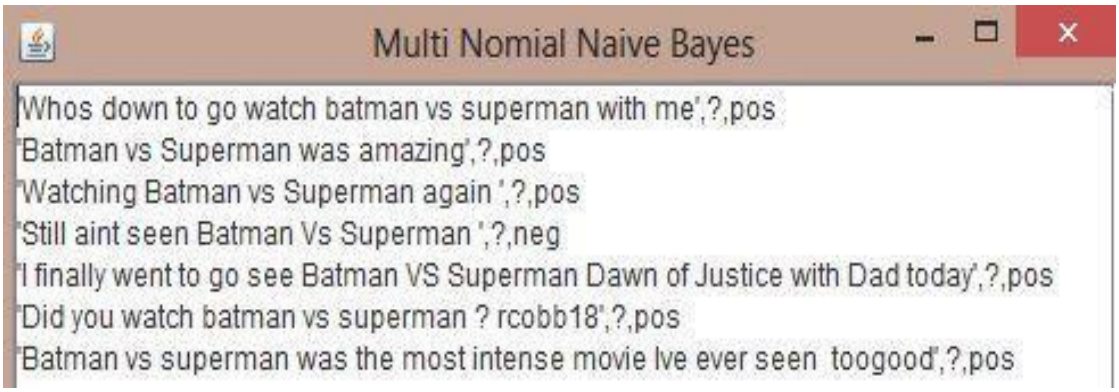


Figure 5.4: Snapshot of results by Multinomial Naïve Bayes Classifier

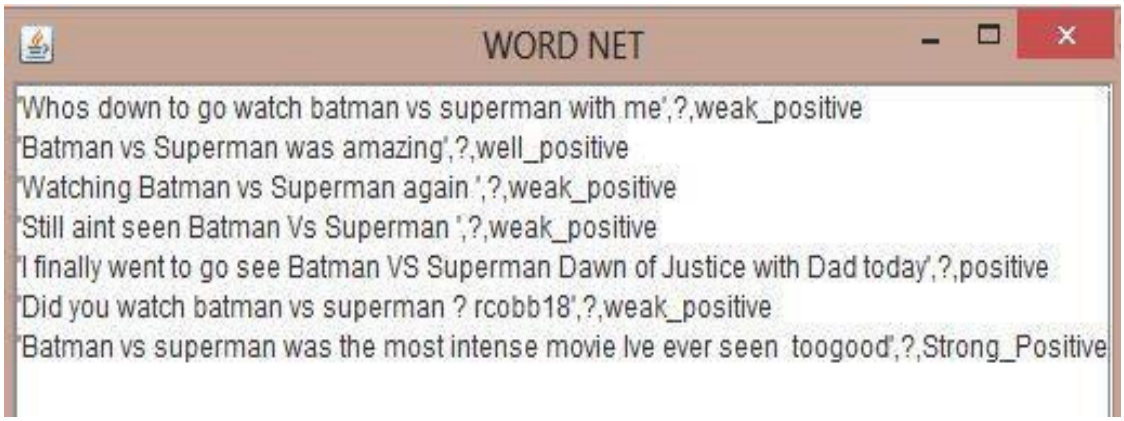


Figure 5.5: Snapshot of results by SentiWordNet

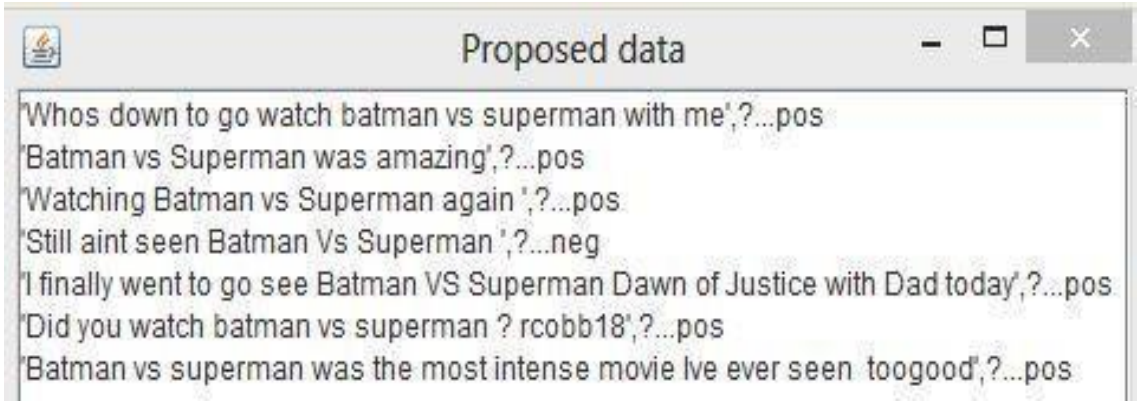


Figure 5.6: Snapshot of Results by proposed algorithm

SUMMARY

For designing and implementing our approach, we designed an application which asks user to input a 'topic' in search textbox. The application uses 'Twitter4J' API to extract the tweets. These tweets are saved in Comma Separated Value (CSV) and are then preprocessed according to methods discussed above. CSV file is converted to Attribute Relation File Format (ARFF) to match the training dataset. This is now our 'test set'. We then use WEKA API to train classifier using training dataset. We used the labelled dataset available online on this link [20] here we choose n-grams and possible features. This creates a classifier model. The test set is then fed to the model to find out the class. Using same application, if user wants to get results using Semantic approach. The same CSV file is then used to get sentiment scores using POS tagging and SentiWordNet algorithm. The POS tagger used is Stanford POS tagger, which is available online. The user then goes on to find the sentiments of the same topic using combined approach as

discussed above and get the results, following which the results are assessed using Precision, Recall and F-Measure.

RESULTS

INTRODUCTION

Results are extracted in two ways. Once the test is taken from the training dataset and results are analysed and second the data is extracted in real time and results are shown in section 6.2.1 and 6.2.2.

RESULTS

The results are assessed using Precision, Recall and F-Measure using following formulae:

$$= \frac{}{+} \quad (6.1)$$

$$= \frac{}{+} \quad (6.2)$$

$$_ = 2 \cdot \frac{}{+} \quad (6.3)$$

$$= \frac{}{+++} \quad (6.4)$$

Here, is True Positive, is False Positive, is True Negative and is False Negative. The results are analysed on two different Test-set. First analysis is done in a conventional way using test-set from a part of training dataset, results of which are shown in table 6.1 and analysed in graph 6.1. In the table 6.1, on one column there is 'No. of tweets' in increasing order and in other columns are results. In graph 6.1, on xaxis, there is 'No. of tweets' in increasing order and on y-axis there is value of results between 0 and 1.

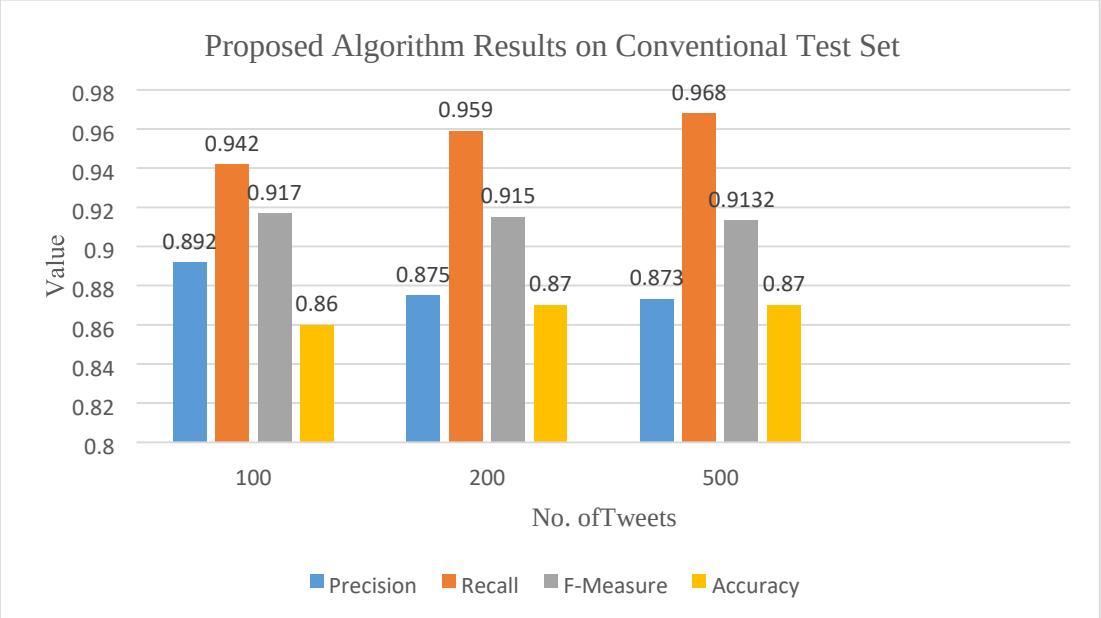
Case Study for offline data: Data is collected from the test set, which is a part of training dataset (available offline) When 100 tweets are taken into consideration, we find out that $T_p = 66$, $T_n = 24$, $F_p = 8$ and $F_n = 4$. Therefore value of precision, recall, fmeasure and accuracy is 0.892, 0.942, 0.917 and 0.86 from equations (6.1), (6.2), (6.3)

and (6.4). While taking 200 tweets into consideration, T_p = 138, T_n = 36, F_p = 20 and F_n= 6 and value of precision, recall, f-measure and accuracy is 0.873, 0.958, 0.9132 and 0.87 from the same equations. Likewise when 500 tweets are taken into consideration, value of precision, recall, f-measure and accuracy comes out to 0.873, 0.968, 0.9132 and 0.87.

Table 6.1: Results of proposed algorithm on conventional test-set

No. of Tweets	Precision	Recall	F-Measure	Accuracy
100	0.892	0.942	0.917	0.86
200	0.875	0.959	0.915	0.87
500	0.873	0.968	0.9132	0.87

Graph 6.1: Coherent view of results of proposed algorithm



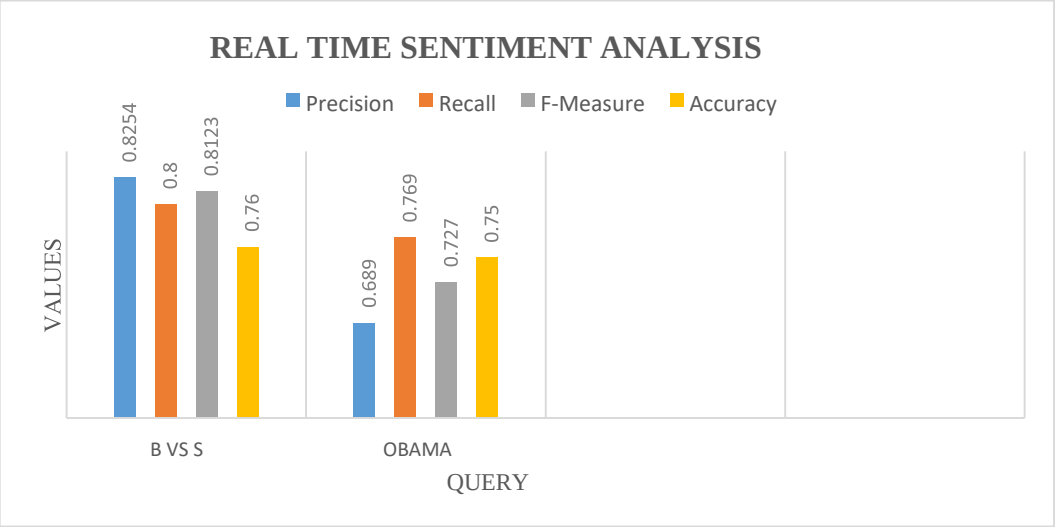
It is evident from the graph that, when the number of tweets are increased from 100 to 200 to 500, the value of accuracy comes out to 0.86, 0.87 and 0.87 respectively. Thus giving an idea that proposed system fairs out quite good with increasing scale.

Case Study for Real Time Data: For Real time data, for search query ‘Batman vs Superman’, number of extracted tweets were 100.

Table 6.2: Results from real-time data

Query	Precision	Recall	F-Measure	Accuracy
B vs S	0.8254	0.8	0.8123	0.76
Obama	0.689	0.769	0.727	0.75

Graph 6.2: Results of proposed algorithm on Real Time tweets



Results assessed on test set from real-time data, using Precision, Recall and F-Measure are shown in table 6.2 and graph 6.2. In the table 6.2, queries are shown in the first column and the results are shown in the other columns. In the graph 6.2, on x-axis is the queries put in as input and on the y-axis is the values between 0 and 1. After classification and on analysis we find out that T_p = 52, T_n = 24, F_p = 11 and F_n= 13. There value of precision comes out to 0.8254 from equation (6.1), value of recall is 0.8 from equation (6.2), value of F-Measure is 0.8123 from equation (6.3), and accuracy is 0.76 from equation (6.4).

SUMMARY

The performance is analysed on the basis of the available dataset as well as real time data. Since the aim is to provide best possible means for sentimental analysis on real time tweets, Multinomial Naïve Bayes algorithm is chosen among other classifier as it gives most 85.06% accuracy and is also faster than other sophisticated algorithms. Also unigram model is chosen as it performs best on the available dataset. To make the Sentiment Analysis more accurate, the machine learning algorithm is combined with

Semantic analysis. Therefore, the MNB classifier is followed with SentiWordNet algorithm which increases the accuracy to 86%. Testing the proposed algorithm on Real Time data, gives up to 77% accuracy.

CONCLUSION 41

CONCLUSION

The objectives of the proposed research work has been stated out clearly. Existing work that is available that has been the inspiration in identification of problem, which was helpful in realisation of this proposed research has been discussed in the problem identification. Along with this, the research methodology used is briefly discussed. The research methodology discusses briefly the methodologies, process and tools used in making of this research. The process of Sentiment Analysis and classification techniques used to achieve this are also explained. Challenges in Sentiment Analysis along with applications are also briefly discussed.

The proposed system takes in consideration two classification techniques. One being the Machine Learning approach and other being the semantic approach.The set of Machine Learning algorithms along with semantic approach using POS tagging and SentiWordNet to classify the tweets are analysed. Different machine learning algorithms are discussed along with their performance on the basis of performance metrics such as precision, recall, and f-measure. The performance is analysed on the basis of the available dataset as well as real time data. Since the aim is to provide best possible means for sentimental analysis on real time tweets, Multinomial Naïve Bayes algorithm is chosen among other classifier as it gives most 85.06% accuracy and is also faster than other sophisticated algorithms. Also unigram model is chosen as it performs best on the available dataset. To make the Sentiment Analysis more accurate, the machine learning algorithm is combined with Semantic analysis. Therefore, the MNB classifier is followed with SentiWordNet algorithm which increases the accuracy to 86%. Testing the proposed algorithm on Real Time data, gives up to 77% accuracy. Challenges that are met are also discussed.

FUTURE WORK

Although authors have gone length to discuss, address and conceptualize the research objectives, but there are few challenges that can be incorporated in the future research work. They are discussed as below:

a) Dealing with Context Related Problem

Tweets that are extracted are spot on, they may not resemble to the training dataset. This affects the performance of the classifier. The major challenge that was faced during sentiment analysis of real time tweets was labelling the tweets with the context. If a user wants to do sentiment analysis of a movie say 'X'. Then it is possible to get tweets which implies different sentiment to 'X', but convey different overall sentiment of tweet. Example, "Everything failed but the movie, including weather, food and the ride. Bad experience" Now it is relevant to stick to either overall sentiment of tweet or just the context of "search query". Future work may address this challenge by asking user if he wants sentiment analysis with the context of search query or on the tweet level.

b) Dealing with blabber

One more challenge is "blabber" which is insignificant chit chat around the subject, which is more of objective in nature and cannot be put against any class. These can actually be dealt by keeping the 'neutral' class. Future work will involve neutral class to handle such objective tweets.

c) Multithreading

Since the research objective is to give real time analysis of sentiments of tweets, therefore multithreading can vastly enhance the results and become useful in realization of effective real time sentiment analysis.

REFERENCES

1) Yessenalina A, Yue Y, Cardie C, "Multi-level Structured Models for Document-level Sentiment Classification" in Proceedings of the 2010 Conference on Empirical Methods

in Natural Language Processing, pages 1046–1056, MIT, Massachusetts, USA, 9-11 October 2010.

2) Hatzivassiloglou V, McKeown K," Predicting the Semantic Orientation of Adjectives", 1997. B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith, "From tweets to polls: Linking text sentiment to public opinion time series," in Proc. 4th Int. AAAI Conf. Weblogs Social Media, Washington, DC, USA, 2010.

3) J.Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," J. Computer Science., vol. 2, no. 1, pp. 1–8, Mar. 2011. G. Mishne and N. Glance, "Predicting movie sales from blogger sentiment," in Proc. AAAI-CAAW, Stanford, CA, USA, 2006

4) Boguslavsky, I. (2017). Semantic Descriptions for a Text Understanding System. In Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue"(2017)

5) Anket, Saleena, N., (2018), An Ensemble Classification System for Twitter SentimentAnalysis, International Conference on Computational Intelligence and Data Science(ICCIDS 2018), Apr 7-8, 2018, The NorthCap University, India