Text Sentiment Analysis Using Natural Language Processing

Amit Kumar Tyagi [0000-0003-2657-8700]
Department of Fashion Technology, National Institute of Fashion Technology, New Delhi, Delhi India
amitkrtyagi025@gmail.com

Shyam Mohan J S
Department of CSE, GITAM University Bengaluru.
sjayaram@gitam.edu

Akshita Tyagi
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, 600127, Tamilnadu, India.
akshitatyagi222@gmail.com

ABSTRACT. In this paper, we discuss our basic text sentiment analysis attempts. The goal of this research is to use participants to extract sentiment from text. It determines sentiment relating to a specific topic using Natural Language Processing algorithms. Subjectivity classification, semantic association, and polarity classification are the three fundamental steps in our effort to categories sentiment. The experiment takes advantage of sentiment lexicons by constructing a syntactic relationship between them and the topic. A substantial annotated data set exists in which a statement was manually classified beyond the six fundamental emotions: joy, love, rage, fear, surprise and sadness. Using the annotated data set, create an emotion vector for the key word in the input sentence. Calculate the emotion vector using an algorithm. of a phrase from the emotion vector of a word. The sentence was then classified into relevant emotion classes based on the emotion vector. In comparison to an individual method, the results are demonstrated and determined to be satisfactory. The goal of this article is to showcase many of the most significant text document categorization approaches and methodologies, as well as to raise awareness of some of the intriguing difficulties that remain unanswered, particularly in the fields of machine learning techniques and text representation.

KEYWORDS- Sentiment Analysis, Natural Language Processing, Machine Learning, Deep Learning

I. INTRODUCTION

Twitter (Kaggle.com) is a popular microblogging platform that lets users to transmit 140-character messages. Personal ideas or opinions regarding the topics are frequently expressed in these tweets. An approach for obtaining a user's sentiment and opinion from their tweets is sentiment analysis. It is a speedier method of gathering customer input than questionnaires or surveys. It's been looked at automating sentiment extraction from text. For example, Pang and Lee used movie review domains to evaluate sentiment classification methods (Nave Bayes, maximum entropy classification, and Support Vector Machine (SVM)). They were able to get an accuracy of up to 82.9 percent using SVM and the unigram model.

The minimal cuts in graphs approach, presented by Pang and Lee before sentiment classification using a machine learning method, uses just the subjective component of the texts for text categorization. Before classifying the feeling as good or negative, they classed the content as sentimental. With an accuracy of 86.4 percent, it outperformed the previous experiment. In addition to machine learning techniques, natural language processing (NLP) methodologies have been described. NLP classifies the polarity of sentiment lexicons and defines a given issue's sentiment expression.

Instead of identifying the sentiment of the entire text based on the specific subject, NLP may identify the text fragment using subject and sentiment lexicons to do sentiment classification. One of the NLP approaches is the feature extraction algorithm. It may be extracted subject-specific characteristics, extracted sentiment from each sentiment-bearing lexicon, and each sentiment's relationship to a certain subject. It beat machine learning methods, with 87 percent accuracy for online review articles and 919 percent accuracy for evaluations of ordinary web pages and news items. This method concentrated on generic text and eliminated several challenging circumstances in order to get better results, such as confusing sentences or sentences with no feeling. This study proposes a method for doing Twitter sentiment analysis based on a specific topic. Many pre-processing processes were used to remove noise from tweets and display them in formal language. NLP is used to determine the sentiment of tweets by identifying the subjective element of tweets that is associated with the subject and classifying the sentiment of tweets. Positive, negative, and neutral labels will be assigned to the text.

This research examines related literature released between 2014 and 2019 to gain a better knowledge of sentiment analysis' applicability in social media text platforms. Analyzing public opinion is a technique that employs Natural Language Processing (NLP) to Extract, transform, and analyses sentiment in a text to classify it as positive, negative, or natural [4]. The majority of past research used sentiment analysis to better understand their customers and make the appropriate decisions to improve their products or services [5].

II. OVERVIEW OF FRAMEWORK

The suggested system's outline is presented in this section. For this experiment, text was taken from kaggle.com, an internet website database. Positive, negative, and neutral labels were manually applied to each dataset. Some preprocessing was done on the dataset before the suggested system investigated it further to give the text in an ordered way Pre-processing guarantees that tweets are written in a formal expressions structure that machines can read and understand. After pre-processing, sentiment classification may be used to identify the sentiment of tweets. The three procedures in sentiment classification are subjectivity classification, semantic association, and polarity classification. By identifying the sentiment using the method Emotion detection algorithm, the sentiment classification projected the text as positive, negative, or neutral. Sentiment analysis may be used to world events such as athletic events, sporting activities, and natural disasters [8,15].

Dataset

Nave Bayes, Decision Tree, and Support Vector Machines were the machine learning methods used. The Emotion Detection method is used in this study. We used the formula with and without NLTK in this method. We manually tokenize the words and remove those that have no meaning without NLTK. However, with NLTK, we utilize libraries that are included in NLTK, so all of the work of tokenizing and deleting undesired terms is handled by the packages included in NLTK, all we have to do is import them into our project.

III. PROPOSED SYSTEM

The suggested system's processes are depicted in the diagram below, starting with pre-processing and ending with sentiment categorization. Section 1 of preprocess discusses the pre-processing procedures in depth, whereas Section 2 of sentiment classification explains the sentiment categorization flow refer figure 1 for this.

1) Pre-process

Because most tweets are unstructured text, pre-processing seeks to arrange and display the tweets in an ordered way, as well as improve machine understanding of the material. URLs and hashtags are removed, special symbols are replaced, repetitive letters are removed, abbreviations and acronyms are expanded, and the topic is capitalized.

To minimize misunderstanding during text processing, special symbols are replaced with words, such as '>' for ‘greater' for 'greater' and '&' for 'and'. Word-based analysis beat emoticon-based analysis for emoticons, according to research on Automatic Sentiment Analysis of Twitter Messages [7]. As a result, emoticons have been deleted from tweets.

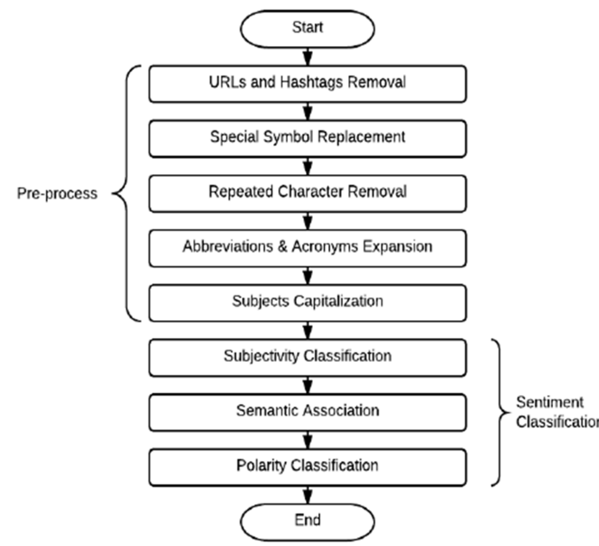

Fig1. Pre-process and Sentiment Classification Flow Chart

2) Sentiment Classification

a) Subjectivity Classification

The tweet will be regarded as subjective based on its weightage. Otherwise, it will be objective, and hence neutral.

“*Come and get internet package*”

or

“*Come and get new internet package*”

In the initial tweet, there is no phrase with an emotion score. It will be labelled Neutral and placed in the Objective category. The word "new" receives an emotion score in the second tweet. Before proceeding on to the semantic association phase, the tweet will be identified as subjective.Texts [1] provide straightforward opinion and comparisons. In direct opinion, sentiment lexicons use prepositions and conjunctions to describe one or more subjects. While there are at least two topics in comparative opinion, the subjects are linked to the same Without the use of a conjunction, sentiment lexicons are created. “I love Unifi," for A good example of straightforward opinion is. As indicated in the example, 'I' is the nominal subject of 'love,' while 'Unifi' is the direct object of 'love’. It's a straightforward assertion.

b) Associative Semantics

As seen in the example, 'I' is the nominal subject of 'love,' whereas 'Unifi' is the direct 'Love's object'. It's a simple statement to make. As previously stated, the majority of grammatical relationships indicate that verbs and adjectives are linked to the subject. We must examine the attribute of 'love' in this case because the Subject 'Unifi' is the direct object of 'love.' We can see that 'I' is a 'Love' is a verb, while 'Unifi' is a noun, according to the POS tag result. Because 'love' is a verb that has a subject, it will be subjected to polarity categorization to see if it has a positive or negative connotation. 'I' As seen in the example, is the nominal subject of 'love,' but 'Unifi' is the direct object of 'love'. It's a straightforward assertion. The bulk of grammatical connections imply that verbs and adjectives are connected to the subject, as previously indicated. We must examine the attribute of 'love' in this case because the Subject 'Unifi' is the direct object of 'love.' From the POS tag result, we can see that 'I' is a personal pronoun, 'love' is a verb, and 'Unifi' is a noun. Because 'love' is a verb that has a subject, it will be subjected to polarity categorization to see if it has a positive or negative connotation.

c) Classification of Polarity

The sentiment expressed in 'I adore Unifi,' for example, is the verb 'love.' The word 'love' has a positive score of 0.625 according to SentiWordNet. As a result, we can conclude that 'Unifi' has a positive attitude, and the sentence should be classified as such. In a nutshell, text polarity refers to the degree to which a piece of text is negative or positive. The entire blend of positive and negative emotions in a statement is measured by polarity. This is notoriously difficult for computers to anticipate. The subject's position is critical for comparative opinion [3].

3) NLP Emotion Detection Algorithm

Steps in volved in this algorithm are:

1. Check if the word in final word list is also present in the emotion.txt file in which the text is there which is used to analyze.
2. Open the emotion file.
3. Loop through each line and clear it.
4. Extract the word and emotion using split.
5. If word is present ->Add the emotions to emotion list.
6. Finally count each emotion in the emotion list.

4) Applications:

Few applications can be included here as:

A) Support in decision making: Making decisions is a crucial aspect of our lives. Opinions gleaned from reviews assist us in making selections such as "which books to buy," "which hotel to stay at," "which movie to watch," and so on.

B) Business application: Making decisions is a crucial aspect of our lives. Opinions gleaned from reviews assist us in making selections such as "which books to buy," "which hotel to stay at," "which movie to watch," and so on.

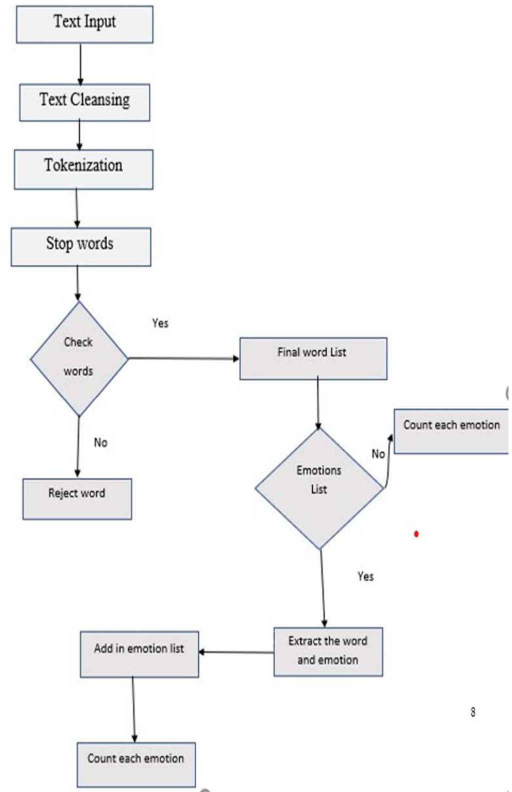

Fig2. Block Diagram of Emotion Detection Algorithm

C) Predictions and trend analysis: By tracking public sentiment, sentiment analysis allows one to forecast market trends. It's especially useful during elections when candidates want to know what people expect from them.

Here, the block diagram shows the step involves in classification refer figure 2 for this.

5) Methodology:

A) Feature selection: The initial step in sentiment classification is to extract characteristics from text that are relevant to the sentiment and that are:

1. **POS tagging-** It's a means of describing a word in a content (corpus) in terms of parts of speech, based on both its definition and its proximity to other words. Nouns, pronouns, adjectives, adverbs, and other components of speech are examples. The majority of the feelings in text are held by adjectives and adverbs.
2. **Stemming-**It is the removal of prefixes and suffixes from a word. 'Playing' and 'played', for example, can be stemmed to 'play'. It aids categorization, although it can occasionally reduce classification accuracy.
3. **Stop words-**Stop words include pronouns (he/she, it), articles (a, the), and prepositions (in, close, alongside). They give little or no information regarding feelings. A list of stop words may be found on the internet. It can be used in the pre-processing phase to eliminate them.

4. **Conjunction handling-** Each sentence, on average, communicates just one meaning at a time. Although, terms like but, while however, and however shift the meaning of the statement completely. For example, while the movie was enjoyable, it fell short of my expectations. Throughput may be enhanced by 5% by following these principles.
5. **Negation handling-**Negative words, such as "not," invert the meaning of the entire phrase. The word 'good' in the phrase the movie was not good is positive, but the word 'not' inverts the polarity to negative.

In this emotion detection algorithm, we use NLTK python library which is a trademark for predicting sentiment analysis and it is a multi-language supporter. In this we are going to be using this analytical library to tokenize. We split the sentence into words and also remove stop words from a read.txt file. So, we tokenize by importing the analytical laboratory are just writing from nltk.tokenize we are importing this something known as the word tokenizer and then we convert all the text into lower case and remove all the punctuation mark from the text. For removing the stop words which containing no meaning by importing something else from NLTK like nltk.corpus So, corpus is basically the data that NLTK has and from that we are just going to import the stop words, so it has a list of stop words inside its corpus data set. Corpus is just like a English word for like a data set but anyways now that we are imported all the stop words, we're just gonna remove all the stop words from our final words from the tokenized words. So, by this way we cleaned our text and match all those words with all the emotions by using vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer This will give us the output that how many words are positive and negative or neutral with the graph itself and for this wehave to install matplotlib. So, this is the way this algorithm works figure 3 shows the frequency and attributes of sentiments in the graph below.

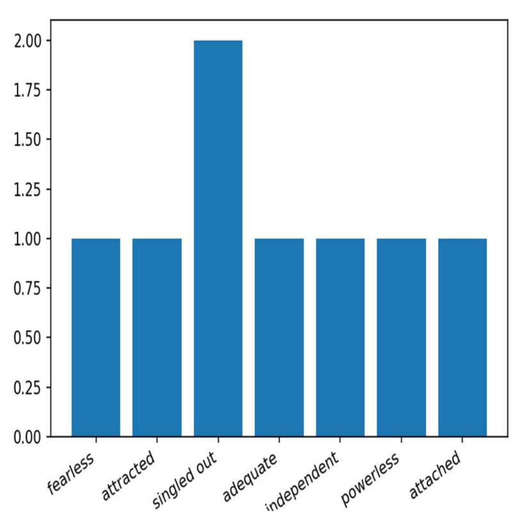

Fig.3 Graph with frequency showing the emotion intensity

.

IV. MACHINE LEARNING TECHNIQUES

Unsupervised, supervised, and semi-supervised approaches can all be used to classify texts. Many strategies and algorithms for clustering and categorization of electronic documents have recently been suggested. Using the available literature, this part focuses on supervised classification approaches, new advancements, and some of the potential and obstacles. As the internet usage rate has rapidly increased, the automatic categorization of content into predetermined categories has been noted as an active focus. Machine learning approaches [33] such as Bayesian classifiers, Decision Trees, K-nearest neighbor (KNN), Support Vector Machines (SVMs), Neural Networks, Latent Semantic Analysis, Rocchio's Algorithm, Fuzzy Correlation, and Genetic Algorithms, among others, have been extensively studied in the last few years, and rapid progress appears in this area (refer figure 4 to figure 9).

A. Rocchio's Algorithm

The Rocchio's Algorithm is a vector space method for document routing or filtering in information retrieval. It builds prototype vectors for each class using a training set of documents, i.e., the average vector over all training document vectors that belong to class ci, and calculates similarity between test document and each of prototype vectors, assigning test document to the class with the highest similarity.

$$C_i = \alpha * centroid_{C_i} - \beta * centroid_{\overline{C_i}}$$

When a category is specified, the vectors of documents that correspond to that category are given a positive weight, while the vectors of the remainder papers are given a negative weight. The prototype vector of this category is created by combining the positively and negatively weighted vectors.

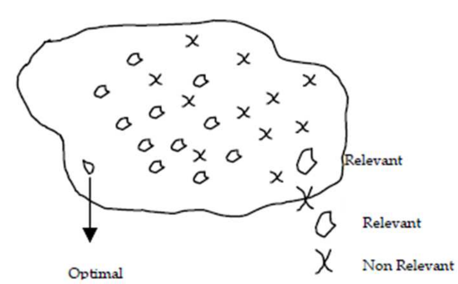

Fig.4. Rocchio Best query for distinguishing between important and irrelevant documents

This method [16] is simple to construct, computationally efficient, quick to train, and has a relevant feedback mechanism, however it has a low classification accuracy. For classification, linear combination is too simple, and constant and are empirical. This is a popular relevance feedback technique that works with the vector space model [17]. The researchers utilized a variant of Rocchio's technique in a machine learning setting, i.e., for generating a user profile from unstructured text [18, 19]. The objective in these applications is to automatically induce a text classifier that can discriminate between document classes.

B. K-nearest neighbor (k-NN)

The k-nearest neighbour method (k-NN) [20] is used to determine the category of test documents by testing the degree of similarity between documents and k training data and storing a specified quantity of classification data.This approach is an instant-based learning algorithm that classified objects based on the training set's nearest feature space [21]. The training sets are represented as a multi-dimensional feature space. The feature space is divided into regions based on the training set's categorization.

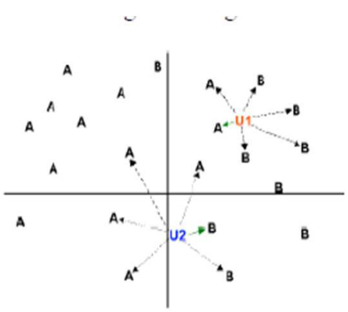

Fig.5.K-nearest neighbor Diagram

This approach is efficient, non-parametric, and simple to use. In comparison to the Rocchio method, more local properties of documents are examined, but the classification time is longer and finding the ideal value of k is challenging. i.e., to assess the k-NN and the Rocchio algorithm, certain flaws in each are highlighted in [22].

C. Decision Tree

By generating well-defined true/false queries in the form of a tree structure, the decision tree rebuilds the manual classification of training papers. The leaves of a decision tree structure reflect the associated document category, while the branches represent the characteristics that lead to those categories. By placing a document in the root node of the tree and allowing it to travel through the query structure until it reaches a certain leaf, which reflects the document's classification aim, a well-organized decision tree can readily categorize it.

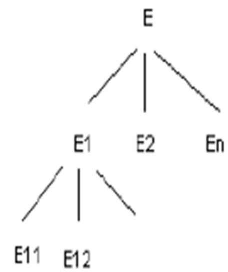

Fig.6. Decision Tree Diagram

With various advantages, the decision tree categorization approach stands out among other decision assistance tools.

The fundamental benefit of a decision tree is that it is easy to comprehend and interpret, especially for non-expert users. Furthermore, using basic mathematical methods, the explanation of a given result may be simply repeated, providing a consolidated perspective of the classification logic, which is important classification information.

D. Decision Rules Classification

To categorize documents into their annotated categories, the decision rules classification approach employs rule-based inference [23]. The algorithms provide a set of rules that characterize each category's profile. Typically, rules are written in the format "IF condition THEN conclusion," with the condition section filled with category attributes and the conclusion portion representing the category's name or another rule to be evaluated. The rule set for a certain category is then built by merging all of the rules from the same category using logical operators such as "and" and "or." Not every rule in the rule set needs to be met throughout the categorization jobs. When dealing with a dataset with a large number of features for each category, using heuristics to decrease the size of the rules set without impacting classification performance is advised. In [24], author proposes a spam filtering system that combines rule-based processing and back-propagation neural networks. Instead of employing keywords, the spamming behaviours are used as characteristics to describe emails in this study.

E. Naïve Bayes Algorithm

The naive Bayes classifier is a simple probabilistic classifier that uses Bayes' Theorem and strong independence assumptions to classify data. Independent feature model is a more descriptive phrase for the underlying probability model. Because of these feature independence assumptions, the sequence of features is immaterial, and the presence of one feature does not impact the presence of other features in classification tasks [25]. These assumptions make the Bayesian classification approach's calculation more efficient, but they significantly limit its usefulness. The nave Bayes classifiers may be taught extremely efficiently, depending on the precise form of the probability model, by using a relatively minimal amount of training data to estimate the parameters essential for classification. Also, the nave Bayes classifier has the benefit of just requiring a limited quantity of training data to estimate the classification parameters. As long as the proper category is more likely than the others, the Bayesian classification strategy will arrive at the correct categorization. The odds of a category do not need to be extremely accurately calculated. To put it another way, the entire classifier is resilient enough to overlook major flaws in its underlying naive probability model.

The fundamental downside of the nave Bayes classification technique is its low classification performance when compared to other discriminative algorithms, such as the SVM, which outperforms the nave Bayes classification approach in terms of classification efficacy. As a result, several active studies have been conducted to determine why the naive Bayes classifier fails in classification tasks and to improve traditional methods by including some useful and efficient methodologies [26][27].

$$P(c_i \mid D) = \frac{P(c_i)P(D \mid c_i)}{P(D)}$$

$$P(D \mid c_i) = \prod_{j=1}^{n} P(d_j \mid c_i)$$

Where P(Ci)= $P(C = c_i) = \frac{Ni}{N}$

and P(dj|ci) = $P(d_j \mid c_i) = \frac{1 + N_{ji}}{M + \sum_{k=1}^{M} N_{ki}}$

Fig.7. Formula to calculate Nave Bayes

For many years, Nave Bayes has been one of the most prominent machine learning algorithms. Its simplicity makes the framework appealing in a variety of tasks, and respectable results are produced in the tasks, despite the fact that this learning is based on an incorrect assumption of independence. As a result, there have been numerous fascinating investigations into naive Bayes.

F. Artificial Neural Network

Artificial neural networks are made up of a vast number of parts with input fans that are orders of magnitude bigger than standard designs' computational elements [28]. These parts, especially artificial neurons, are linked together in a group utilizing a mathematical model for data processing based on a connectionist approach to computing. The neurons in neural networks are sensitive to the items they store. It may be used to store a large number of instances represented by high-dimensional vectors in a distortion-tolerant manner.

Document categorization tasks have been implemented. Due to its ease of implementation, several studies employ the single-layer perceptron, which consists of simply an input layer and an output layer. Through a succession of weights, inputs are fed directly to outputs. It may be regarded the simplest type of feed-forward network in this sense. The more advanced multi-layer perceptron, which has an input layer, one or more hidden layers, and an output layer in its structure, is also commonly used for classification applications [28].

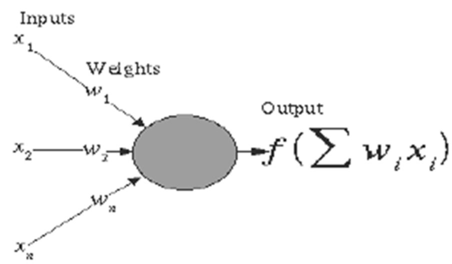

Fig.8. Artificial Neural Network

The ANN has the ability to get pre-synaptic connections provide inputs xi, synaptic effectiveness is modelled using actual weights wi, and the neuron's response is a nonlinear function f of its weighted inputs. O_{pj} is the output of neuron j for pattern p.

$$O_{pj}(net_j) = \frac{1}{1 + e^{-\lambda \, net_j}}$$

and

$$net_j = bias * W_{bias} + \sum_k O_{pk} W_{jk}$$

For document categorization, a neural network produces good results. Outcomes in complicated domains and may be used for both discrete and continuous data and continuous data (which is especially useful for continuous data domain). Testing is quick, but training takes a long time. Users find it challenging to evaluate sluggish and learnt outcomes. It is compared to taught rules (comparing with Decision tree), Empirical Risk Minimization (ERM) motivates ANN to strive to reduce risk as much as possible. Note that Overfitting may occur if training error is minimized.

G. Support Vector Machines

Support vector machines (SVMs) are a type of discriminative classification algorithm that is widely acknowledged as being more accurate. The SVM classification approach is based on the computational learning theory idea of structural risk minimization [109]. This principle's goal is to develop a hypothesis that guarantees the smallest true error. Furthermore, the SVM are well-founded and theoretically understandable and analyzable [29].

Other classification algorithms do not require both positive and negative training sets, like the SVM does. These positive and negative training sets are required for the SVM to find the decision surface in n-dimensional space, also known as the hyper plane, that best separates positive and negative data. The support vector refers to the document representations who are closest to the decision surface. When documents that do not correspond to the support vectors are deleted from the training data set, the SVM classification performance stays unaltered.

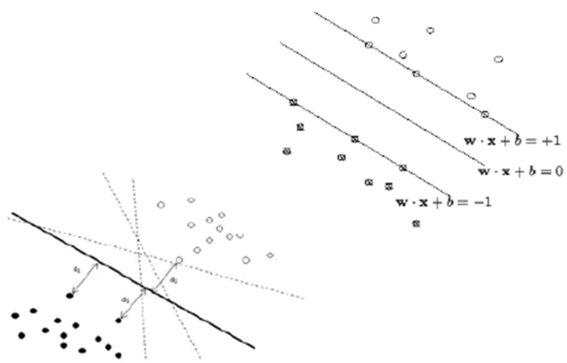

Fig.9. The best separating hyper plane, hyper planes, and support vectors are shown in this diagram.

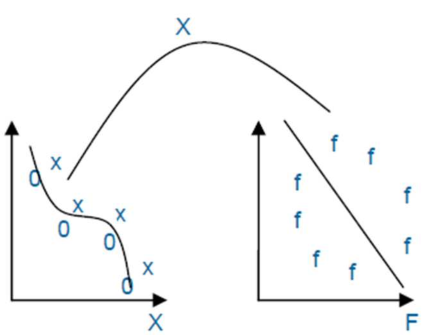

Fig.10. Non-linear input space to high-dimensional space mapping.

With its remarkable classification performance, the SVM classification approach stands out from the others [30, 31 and 32]. Furthermore, it can handle documents with a large input area and removes the majority of the non-essential components. The main disadvantage of SVMs is their relatively sophisticated training and categorization algorithms, as well as their high time and memory consumption during the training and classification stages. Furthermore, confusions arise during classification tasks since documents may be assigned to many categories because similarity is often assessed separately for each group.

As a result, SVM is a supervised learning approach for classification that identifies the linear separating hyperplane that maximises the margin, i.e., the Optimum Separating Hyperplane (OSH), and maximises the margin between the two data sets. Two parallel hyperplanes, one on either side of the separating hyperplane, are built and "pushed up against" the two data sets to determine the margin.

Intuitively, the hyperplane with the greatest distance to the nearby data points of both classes achieves a decent separation, because the higher the margin, the smaller the classifier's generalization error. Increasing the profit margin is the same as increasing the profit margin.

$$\underset{w\,b\,\zeta_i}{\text{minimize}} \quad \frac{1}{2}w^T w + C\left(\sum_{i=1}^{N} \zeta_i\right)$$

$$\text{subject to} \quad y_i(w^T x_i - b) + \zeta_i - 1 \geq 0, \quad 1 \leq i \leq N$$

$$\zeta_i \geq 0, \quad 1 \leq i \leq N$$

Introducing Lagrange multipliers α, β, the Lagrangian is:

$$\begin{aligned}\ell(w,b,\zeta_i;\alpha,\beta) &= \frac{1}{2}w^T w + C\sum_{i=1}^{N}\zeta_i \\ &\quad - \sum_{i=1}^{N}\alpha_i[y_i(w^T x_i - b) + \zeta_i - 1] - \sum_{i=1}^{N}\mu_i\zeta_i \\ &= \frac{1}{2}w^T w + \sum_{i=1}^{N}(C - \alpha_i - \mu_i)\zeta_i \\ &\quad - \left(\sum_{i=1}^{N}\alpha_i y_i x_i^T\right)w - \left(\sum_{i=1}^{N}\alpha_i y_i\right)b + \sum_{i=1}^{N}\alpha_i\end{aligned}$$

Multiple optimum techniques, such as a unique significance weight definition, feature selection utilizing the entropy weighting scheme, and optimal parameter values, are used to construct an optimal SVM algorithm. The SVM is the most effective tool for document classification. In the last, researchers can refer several useful articles on machine learning or artificial intelligence and their applications in different sectors/ domains in [34-60].

V. CONCLUSION AND FUTURE SCOPE

There is a lot of research on measuring sentiment from text because Twitter, kaggle.com, and other web portals are popular social media sites Our suggested system, which employs NLP techniques to extract subject from text and determine text polarity by evaluating sentiment lexicons related with subject, is given. The suggested system conducts Emotion Detection Algorithm based on the trials. Future study will concentrate on ways to increase sentiment analysis accuracy. It's also worth noting that, because of the prevalence of misspelt words and slang in texts and tweets, extracting sentiment can be difficult. Lexicons is difficult if the data is not pre-processed to formal language. Tweets should be converted to formal sentences in pre-processing, which is still inefficient because additional training data is required.

REFERENCES

[1] A. Westerski, "Sentiment Analysis: Introduction and the State-of-the-Art overview", Universidad Politecnica de Madrid, Spain, pp 211-218, 2007

[2] S. W. Davenport, S. M. Bergman, J. Z. Bergman, J. Z., and M. E. Fearrington, "Twitter versus Facebook: Exploring the role ofnarcissism in the motives and usage of different social media platforms.", Computers in Human Behavior, 2014.

[3] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis", Computational linguistics 35, no 3, pp 399-433, 2009

[4] Agarwal, Basant, Namita Mittal, Pooja Bansal, and Sonal Garg. (2015) "Sentiment Analysis Using Common-Sense and Context Information." Journal of Computational Intelligence and Neuroscience 9 (2015).

[5] U. T. Gursoy, D. Bulut, and C. Yigit. (2017) "Social Media Mining and Sentiment Analysis for Brand Management." Global Journal of Emerging Trends in e-Business, Marketing and Consumer Psychology 3 (1): 497-551

[6] Sivarajah, Uthayasankar, Muhammad Mustafa Kamal, Zahir Irani, and VishanthWeerakkody. (2017) "Critical Analysis of Big Data Challenges and Analytical Methods".

[7] A. C. Lima, and L. N. de Castro, "Automatic sentiment analysis of Twitter messages", in Computational Aspects of Social Networks (CASoN), 2012 Fourth International Conference, pp. 52-57, IEEE, 2012.

[8] Mahtab, S. Arafin, N. Islam, and M. MahfuzurRahaman. (2018, 21-22 Sept. 2018). "Sentiment Analysis on Bangladesh Cricket with Support Vector Machine", in the 2018 International Conference on Bangla Speech and Language Processing (ICBSLP).

[9] Rahman, S. A. El, F. A. AlOtaibi, and W. A. AlShehri. (2019, 3-4 April 2019). "Sentiment Analysis of Twitter Data", in the 2019 International Conference on Computer and Information Sciences (ICCIS).

[10] Joyce, Brandon, and Jing Deng. (2017) "Sentiment Analysis of Tweets for the 2016 US Presidential Election", in IEEE MIT Undergraduate Research Technology Conference (URTC), Cambridge, MA, USA: IEEE.

[11] Yuliyanti, Siti, Djatna, SukocoTaufik, and Heru. (2017) "Sentiment Mining of Community Development Program Evaluation Based on Social Media." TELKOMNIKA (Telecommunication Computing Electronics and Control) 15 (4): 1858-1864

[12] Martin-Domingo, Luis, Juan Carlos Martin, and Glen Mandsberg. (2019) "social media as a Resource for Sentiment Analysis of Airport Service Quality (ASQ)." Journal of Air Transport Management.

[13] Ragini, J. Rexiline, P. M. Rubesh Anand, and Vidhyacharan Bhaskar. (2018) "Big Data Analytics for Disaster Response and Recovery Through Sentiment Analysis." International Journal of Information Management 42: 13-24.

[14] Ikoro, Victoria, Maria Sharmina, Khaleel Malik, and Riza Batista-Navarro. (2018) "Analyzing Sentiments Expressed on Twitter by UK Energy Company Consumers", in Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS) (pp. 95- 98): IEEE.

[15] Yuliyanti, Siti, Djatna, SukocoTaufik, and Heru. (2017) "Sentiment Mining of Community Development Program Evaluation Based on Social Media." TELKOMNIKA (Telecommunication Computing Electronics and Control) 15 (4): 1858-1864.

[16] Willian W. Cohen and Yoram Singer, "Context-sensitive learning method for text categorization", SIGIR' 96, 19th International Conference on Research and Development in Informational Retrieval, pp-307-315, 1996.

[17] Ittner, D., Lewis, D., Ahn, D; "Text Categorization of Low-Quality Images", In: Symposium on Document Analysis and Information Retrieval, Las Vegas, NV.pp. 301-315, 1995.

[18] Pazzani M., Billsus, D; "Learning and Revising User Profiles", The Identification of Interesting Web Sites. Machine Learning 27(3) pp. 313-331, 1997.

[19] Balabanovic, M., Shoham Y.: FAB; "Content-based, Collaborative Recommendation", Communications of the Association for Computing Machinery 40(3) pp. 66-72, 1997.

[20] Tam, V., Santoso, A., &Setiono, R., "A comparative study of centroid-based, neighborhood-based and statistical approaches for effective document categorization", Proceedings of the 16th International Conference on Pattern Recognition, pp.235–238, 2002.

[21] Eui-Hong (Sam) Han, George Karypis, Vipin Kumar;"Text Categorization Using Weighted Adjusted k-Nearest Neighbor Classification", Department of Computer Science and Engineering. Army HPC Research Centre, University of Minnesota, Minneapolis, USA. 1999.

[22] DuoqianMiao,Qiguo Duan, Hongyun Zhang, Na Jiao, "Rough set-based hybrid algorithm for text classification", Expert Systems with Applications -2009.

[23] ChidanandApte, Fred Damerau, Sholom M. Weiss.; "Towards Language Independent Automated Learning of Text Categorization Models", In Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp. 23-30.1994.

[24] Chih-Hung Wu, "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks", Expert Systems with Applications, pp. 4321–4330, 2009.

[25] Heide Brücher, Gerhard Knolmayer, Marc-André Mittermayer; "Document Classification Methods for Organizing Explicit Knowledge", Research Group Information Engineering, Institute of Information Systems, University of Bern, Engehaldenstrasse 8, CH - 3012 Bern, Switzerland. 2002.

[26] Andrew McCallum, Kamal Nigam; "A Comparison of Event Models for Naïve Bayes Text Classification", Journal of Machine Learning Research 3, pp. 1265-1287. 2003.

[27] Irina Rish, Joseph Hellerstein, Jayram Thathachar; "An Analysis of Data Characteristics that affect Naïve Bayes Performance", IBM T.J. Watson Research Center 30 Saw Mill River Road, Hawthorne, NY 10532, USA. 2001.

[28] Miguel E. Ruiz, Padmini Srinivasan; "Automatic Text Categorization Using Neural Network",In Proceedings of the 8thASIS SIG/CR Workshop on Classification Research, pp. 59-72. 1998.

[29] Thorsten Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant tures" ECML-98, 10thEuropean Conference on Machine Learning, pp. 137-142. 1998.

[30] YiMing Yang, Xin Liu; "A Re-examination of Text Categorization Methods, School of Computer Science", Carnegie Mellon University, 1999.

[31] Tyagi A.K., Fernandez T.F., Mishra S., Kumari S. (2021) Intelligent Automation Systems at the Core of Industry 4.0. In: Abraham A., Piuri V., Gandhi N., Siarry P., Kaklauskas A., Madureira A. (eds) Intelligent Systems Design and Applications. ISDA 2020. Advances in Intelligent Systems and Computing, vol 1351. Springer, Cham. https://doi.org/10.1007/978-3-030-71187-0_1

[32] K. Sekar and A. K. Tyagi, "Study of Data Behaviour and Methods for Data Prediction and Analysis," 2022 6th International Conference on Intelligent Computing and Control Systems

(ICICCS), 2022, pp. 1-6, doi: 10.1109/ICICCS53718.2022.9788360.

[33] Amit Kumar Tyagi, Poonam Chahal, “Artificial Intelligence and Machine Learning Algorithms”, Book: Challenges and Applications for Implementing Machine Learning in Computer Vision, IGI Global, 2020.DOI: 10.4018/978-1-7998-0182-5.ch008

[34] Gillala Rekha, Amit Kumar Tyagi, and V. Krishna Reddy, “A Wide Scale Classification of Class Imbalance Problem and its Solutions: A Systematic Literature Review”, Journal of Computer Science, Vol.15, No. 7, 2019, ISSN Print: 1549-3636, pp. 886-929.

[35] L. Kanuru, A. K. Tyagi, A. S. U, T. F. Fernandez, N. Sreenath and S. Mishra, "Prediction of Pesticides and Fertilizers using Machine Learning and Internet of Things," 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-6, doi: 10.1109/ICCCI50826.2021.9402536.

[36] Amit Kumar Tyagi (2022), Using Multimedia Systems, Tools, and Technologies for Smart Healthcare Services, IGI Global. DOI: 10.4018/978-1-6684-5741-2

[37] Khushboo Tripathi, Manjusha Pandey, and Shekhar Verma. 2011. Comparison of reactive and proactive routing protocols for different mobility conditions in WSN. In Proceedings of the 2011 International Conference on Communication, Computing & Security (ICCCS '11). Association for Computing Machinery, New York, NY, USA, 156–161. https://doi.org/10.1145/1947940.1947974

[38] Jajula, S.K., Tripathi, K., Bajaj, S.B. (2023). Review of Detection of Packets Inspection and Attacks in Network Security. In: Dutta, P., Chakrabarti, S., Bhattacharya, A., Dutta, S., Piuri, V. (eds) Emerging Technologies in Data Mining and Information Security. Lecture Notes in Networks and Systems, vol 491. Springer, Singapore. https://doi.org/10.1007/978-981-19-4193-1_58

[39] Ranchhodbhai P.N, Tripathi K., “Identifying and Improving the Malicious Behavior of Rushing and Blackhole Attacks using Proposed IDSAODV Protocol”, International Journal of Recent Technology and Engineering,vlo. 8(3), pp.6554-6562, 2019

[40] D. Agarwal and K. Tripathi, "A Framework for Structural Damage detection system in automobiles for flexible Insurance claim using IOT and Machine Learning," 2022 International Mobile and Embedded Technology Conference (MECON), 2022, pp. 5-8, doi: 10.1109/MECON53876.2022.9751889.

[41] K. Somisetti, K. Tripathi and J. K. Verma, "Design, Implementation, and Controlling of a Humanoid Robot," *2020 International Conference on Computational Performance Evaluation (ComPE)*, 2020, pp. 831-836, doi: 10.1109/ComPE49325.2020.9200020.

[42] Sai, G.H., Tripathi, K., Tyagi, A.K. (2023). Internet of Things-Based e-Health Care: Key Challenges and Recommended Solutions for Future. In: Singh, P.K., Wierzchoń, S.T., Tanwar, S., Rodrigues, J.J.P.C., Ganzha, M. (eds) Proceedings of Third International Conference on Computing, Communications, and Cyber-Security. Lecture Notes in Networks and Systems, vol 421. Springer, Singapore. https://doi.org/10.1007/978-981-19-1142-2_37

[43] S. Subasree, N.K. Sakthivel, Khushboo Tripathi, Deepshikha Agarwal, Amit Kumar Tyagi, Combining the advantages of radiomic features based feature extraction and hyper parameters tuned RERNN using LOA for breast cancer classification, Biomedical Signal Processing and Control, Volume 72, Part A, 2022, 103354, ISSN 1746-8094, https://doi.org/10.1016/j.bspc.2021.103354.

[44] Kumari, S. &Muthulakshmi, P. (2022). Transformative Effects of Big Data on Advanced Data Analytics: Open Issues and Critical Challenges. Journal of Computer Science, 18(6), 463-479. https://doi.org/10.3844/jcssp.2022.463.479

[45] S. Midha and K. Triptahi, "Extended TLS security and Defensive Algorithm in OpenFlow SDN," *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2019, pp. 141-146, doi: 10.1109/CONFLUENCE.2019.8776607.

[46] Midha, S., Tripathi, K. (2021). Extended Security in Heterogeneous Distributed SDN Architecture. In: Hura, G., Singh, A., Siong Hoe, L. (eds) Advances in Communication and Computational Technology. Lecture Notes in Electrical Engineering, vol 668. Springer, Singapore. https://doi.org/10.1007/978-981-15-5341-7_75

[47] Midha, S., Tripathi, K. (2020). Remotely Triggered Blackhole Routing in SDN for Handling DoS. In: Dutta, M., Krishna, C., Kumar, R., Kalra, M. (eds) Proceedings of International Conference on IoT Inclusive Life (ICIIL 2019), NITTTR Chandigarh, India. Lecture Notes in Networks and Systems, vol 116. Springer, Singapore. https://doi.org/10.1007/978-981-15-3020-3_1

[48] Midha S, Tripathi K, Sharma MK. Practical Implications of Using Dockers on Virtualized SDN. Webology. 2021 Apr; 18, pp.312-30.

[49] S. Midha, G. Kaur and K. Tripathi, "Cloud deep down — SWOT analysis," *2017 2nd International Conference on Telecommunication and Networks (TEL-NET)*, 2017, pp. 1-5, doi: 10.1109/TEL-NET.2017.8343560.

[50] Mapanga, V. Kumar, W. Makondo, T. Kushboo, P. Kadebu and W. Chanda, "Design and implementation of an intrusion detection system using MLP-NN for MANET," *2017 IST-Africa Week Conference (IST-Africa)*, 2017, pp. 1-12, doi: 10.23919/ISTAFRICA.2017.8102374.

[51] Jain and K. Tripathi, "Biometric Signature Authentication Scheme with RNN (BIOSIG_RNN) Machine Learning Approach," *2018 3rd International Conference on Contemporary Computing and Informatics (IC3I)*, 2018, pp. 298-305, doi: 10.1109/IC3I44769.2018.9007284.

[52] Tyagi, A.K. (Ed.). (2021). Data Science and Data Analytics: Opportunities and Challenges (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781003111290

[53] Tyagi, A.K., & Abraham, A. (Eds.). (2022). Recurrent Neural Networks (1st ed.). CRC Press. https://doi.org/10.1201/9781003307822

[54] Tyagi, A.K., & Abraham, A. (Eds.). (2021). Recent Trends in Blockchain for Information Systems Security and Privacy (1st ed.). CRC Press. https://doi.org/10.1201/9781003139737

[55] Kumar Tyagi, A., Abraham, A., Kaklauskas, A., Sreenath, N., Rekha, G., & Malik, S. (Eds.). (2022). Security and Privacy-Preserving Techniques in Wireless Robotics (1st ed.). CRC Press. https://doi.org/10.1201/9781003156406

[56] Tyagi, A. K., Rekha, G., &Sreenath, N. (Eds.). (2021). Opportunities and Challenges for Blockchain Technology in Autonomous Vehicles. IGI Global. http://doi:10.4018/978-1-7998-3295-9

[57] Akshita Tyagi, SwettaKukreja, Meghna Manoj Nair, Amit Kumar Tyagi, Machine Learning: Past, Present and Future, Neuroquantology, Volume 20, No 8 (2022), DOI: 10.14704/nq.2022.20.8.NQ44468

[58] Tyagi, A. K. (Ed.). (2021). Multimedia and Sensory Input for Augmented, Mixed, and Virtual Reality. IGI Global. http://doi:10.4018/978-1-7998-4703-8

[59] Malik, S., Bansal, R., & Tyagi, A. K. (Eds.). (2022). Impact and Role of Digital Technologies in Adolescent Lives. IGI Global. http://doi:10.4018/978-1-7998-8318-0

[60] Kumari S., Vani V., Malik S., Tyagi A.K., Reddy S. (2021) Analysis of Text Mining Tools in Disease Prediction. In: Abraham A., Hanne T., Castillo O., Gandhi N., Nogueira Rios T., Hong TP. (eds) Hybrid Intelligent Systems. HIS 2020. Advances in Intelligent Systems and Computing, vol 1375. Springer, Cham. https://doi.org/10.1007/978-3-030-73050-5_55