Sentiment Analysis of Twitter Data using Voting Classification

[1]Dr. Satyen Parikh, [2]Mitali Shah
[1]Executive Dean, [2]PhD Scholar,
[1]Department of Computer Science
[1]Ganpat University, Kherva, India

Abstract : This research work is carried to perform SA (sentiment analysis) from the text data. The sentiment analysis has various phases which include pre-processing, feature extraction and classification. The SVM classification method is applied for the sentiment analysis. In the second phase, the SVM and KNN classifiers are merged for the classification and tested in five different datasets. In the last phase, the voting classification method is applied for the sentiment analysis which is combination of Gaussian Naive Bayes, Random Forest, KNN and SVM. Python is executed to deploy the existing and suggested methods. Diverse parameters such as accuracy, execution time are considered for analysing the results. The suggested approach provides higher accuracy, pression and recall as compared to SVM model while analysing the sentiments.

Keywords - SVM, Sentiment Analysis, Voting Classifier, Gaussian Naive Bayes, Random Forest

I. Introduction

1.1 Sentiment Analysis

In the past few years, owing the wide-ranging use of social media, sentiment analysis has become popular among huge number of people who have different interests and inspirations. Since users around the world can express their opinion about a variety of topics related to policies, education, travelling, values, commercial goods, or topics of regular life. To derive knowledge from those data has turned out to be a matter of great implication and signification. Knowing their sentiments expressed by their messages across different platforms, apart from information related to the sites visited by users, purchasing preferences, etc., became an important aspect for assessing public opinion regarding a particular topic [1]. A usual way is to categorize the polarity of a text in the context of satisfaction, dissatisfaction or neutrality expressed by a user. Polarity may differ from positive to negative with respect to labelling or number of levels; however, it generally shows the emotions of the text varying from a pleased to sad mode. Figure 1 represents a conceptual model of sentiment analysis.

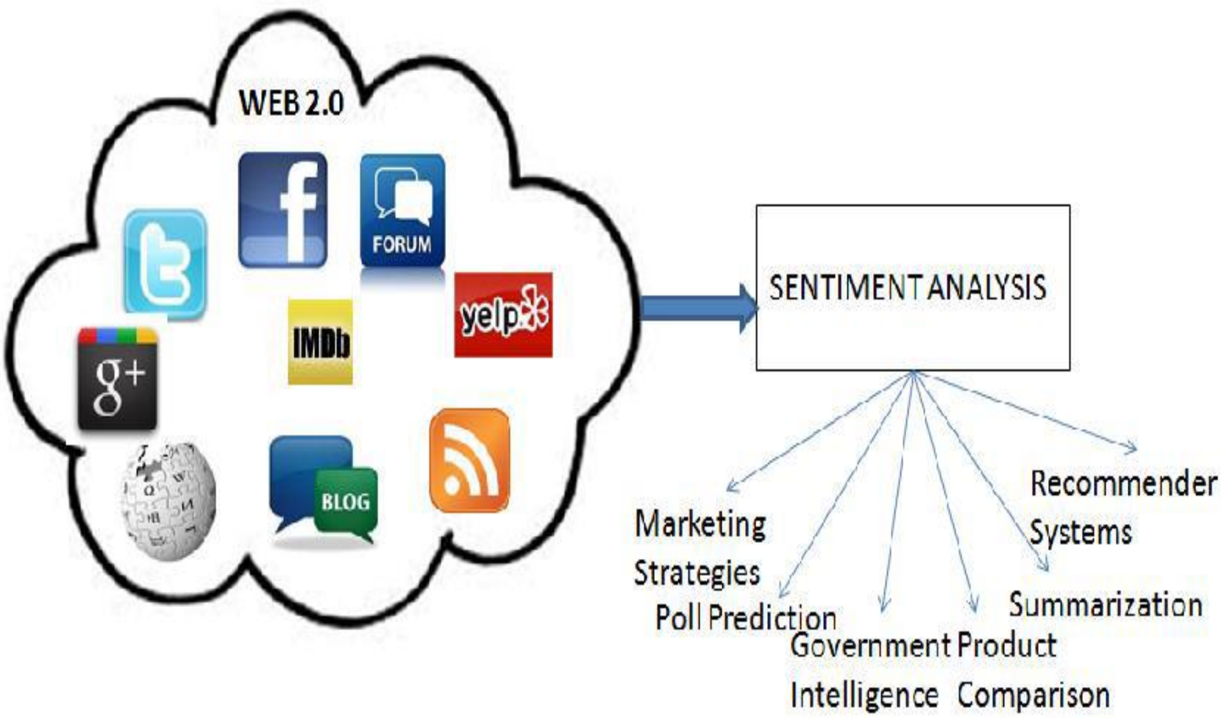

Figure 1: Conceptual model of Sentiment Analysis

Sentiment is a term that not only describes a subject that is subjective and objective but also a factual or non-factual subject that bridges the distinction between a positive or negative subject. Sentiment analysis is an analysis based on the spread of rumours or gossips. Sentiment analysis is an analytical scheme that is used for text analysis. The sentiment analysis aims at determining the subjectivity of opinion [2], the outcome of a review or tweet. Based on sentiment analysis, an individual's opinion can be classified into different classes depending on the data dimensionality and document form. Sentiment analysis, also termed as opinion mining, is an underlying task of NLP natural language processing, and seeks to infer the sentiment polarity of provided texts. In current years, sentiment analysis has efficaciously found wide applications in cookery, e-commerce, hotels and other domains. For instance, the catering sector may opt for sentiment analysis to help customers in finding the food they want from a wide range of restaurants. The new customer can choose more prevalent restaurants on the basis of comments of numerous customers [3].

1.1.1 General Architecture of Sentiment Analysis

Sentiment analysis is generally referred to as a document classification problem, which aims to separate documents expressing positive and negative sentiments. Sentiment analysis adopts the universal structure of a classification problem, which is shown in Figure 2.

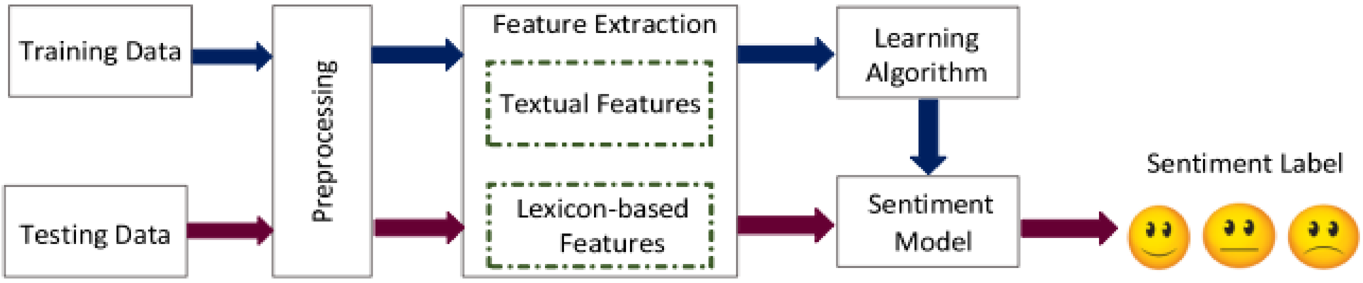

Figure 2: Sentiment Analysis Process

Data Pre-processing is the process applied to clean and prepare text for classification. Online text typically contains a lot of noisy and useless parts, e.g., HTML tags, scripts, and commercials. Moreover, at the level of words, many words in the text have no effect on its general orientation [4]. Placing those words increases the dimensionality of the problem and therefore makes classification more challenging because each word in the text is treated as single dimension. It is assumed that the data is appropriately pre-processed: reducing noise in the text must help enhance the classifier's performance and accelerate the classification process, therefore assist in practical sentiment analysis. Several steps are comprised in the complete process. These steps are online text cleaning, white space removal, abbreviation expansion, stemming, stop words removal, negative handling, and ultimately feature selection. All the steps excluding the last one is related to transformation, while the last step which applies some operations to choose the essential pattern is known as filtering. Characteristics in terms of opinion mining are words, terms or phrases expressing the opinion, as positive or negative in powerful manner. In other words, they greatly affect the text orientation unlike other terms in the similar text. Feature extraction and classifier learning are the main blocks of sentiment analysis framework. Feature extraction involves converting raw text data into demonstrative feature vectors. These features are then fed a learning algorithm to learn the classifier framework [5].

1.1.2 Sentiment Analysis Levels

The pipeline of sentiment analysis comprises three levels to class the sentiment analysis approaches. These levels are document level, sentence level and feature or aspect level [6]. This part briefly reviews various case studies conducted at all levels of sentiment analysis (Figure 3).

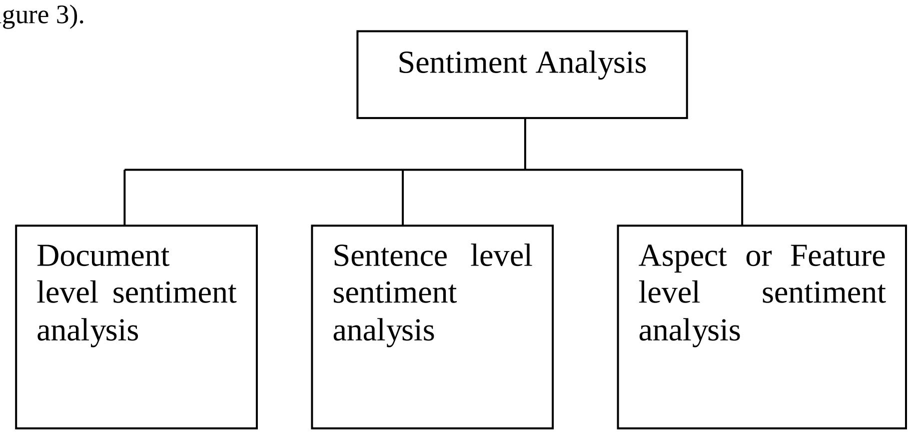

Figure 3: Sentiment Analysis Levels

i. Document level: Document level task aims at classifying whether an overall opinion document conveys a positive or negative sentiment. For instance, after a product is reviewed, the system establishes whether the review declares a complete positive or negative opinion concerning the product. This task is normally referred to as document-level sentiment classification [7]. This level of analysis is based on the hypothesis that every document conveys an opinion on an entity (for example, a product). Therefore, it inappropriate to documents evaluating or comparing multiple objects.

ii. Sentence level: This level of analysis aims at determining whether the opinion conveyed in the sentence is positive, negative or neutral. There are two means to conduct sentence level analysis. One approach is to treat such an analysis as simply a 3-way classification task, where the labels are positive, negative, and neutral. Another approach is to first explore subjectivity in the sentence by dividing opinionated texts from non-opinionated texts, then classifying those subjective texts with either of the two labels (positive or negative). The shortcoming of sentence level analysis is that every sentence is semantically and syntactically linked with other portions of the text. Thus, this task needs both local as well as global circumstantial information [8].

iii. Aspect-level: Both document-level and sentence-level sentiment analysis use fixed sentiment polarity rested upon the entire document/sentence instead of the subjects in the document/sentence. Clearly, this is not appropriate in several cases. Aspect-level sentiment analysis is a sub-function of sentiment analysis, and its goal is to perform sentiment recognition and classification at aspect level [9]. The purpose of aspect-level sentiment analysis is to predict the sentiment polarities of each explicit aspect word in a provided sentence.

1.2 Twitter Sentiment Analysis

These days, Twitter is one of the most influential social networking microblogging services. It provides its services globally as a medium of information sharing. Hence, to extract people opinions from tweets from different topics, measure the impact of various events or classify sentiments became a topic of huge interest. Twitter, with more than413 million monthly active users and more than 500 million tweets a day, has now turned out to be a treasure house for businesses and individuals having a vast social media following in preserving and growing their influence and status. Sentiment analysis provides these businesses potential to monitor various social media sites online. The process of sentiment analysis is to automatically detect whether a part of the text holds sentimental or opinionated content and can further discover the polarity of the text [10]. The purpose of Twitter sentiment classification is to classify the sentiment polarity of a tweet as positive, negative or neutral. Tweets are generally made up of partial, noisy and unstructured sentences, asymmetrical words, misspelled words and non-dictionary words. A tweet contains several unique features that set our research apart from previous research:

- Length: The maximum length of a tweet is 140 characters. From the training set, the average length of a tweet is counted as 14 words or 78 characters. This differs greatly from previous sentiment classification research that has focused on classifying longer bodies of work, such as film reviews.
- Data availability: One more difference is the amount of data existing. The Twitter API makes it extremely easy to gather millions of tweets for training. In previous research, the tests only included thousands of training items.
- Language model: Twitter users use different media to post tweets, such as their smart phones. The frequency of misspellings and slang in Tweet messages is far greater than in other fields.
- Domain: The tweets posted on twitter are of small size. The tweets can be posted about many topics that adapt itself a certain topic, different from other sites. This varies from a big percentage of previous research, which aims at certain fields, e.g., movie reviews [11].

1.2.1 Twitter Sentiment Classification Process

The visual presentation of twitter sentiment analysis process is illustrated in figure 4. The classifier trained by the processed data set is interpreted by three polarities called Positive, negative, and neutral tweets.

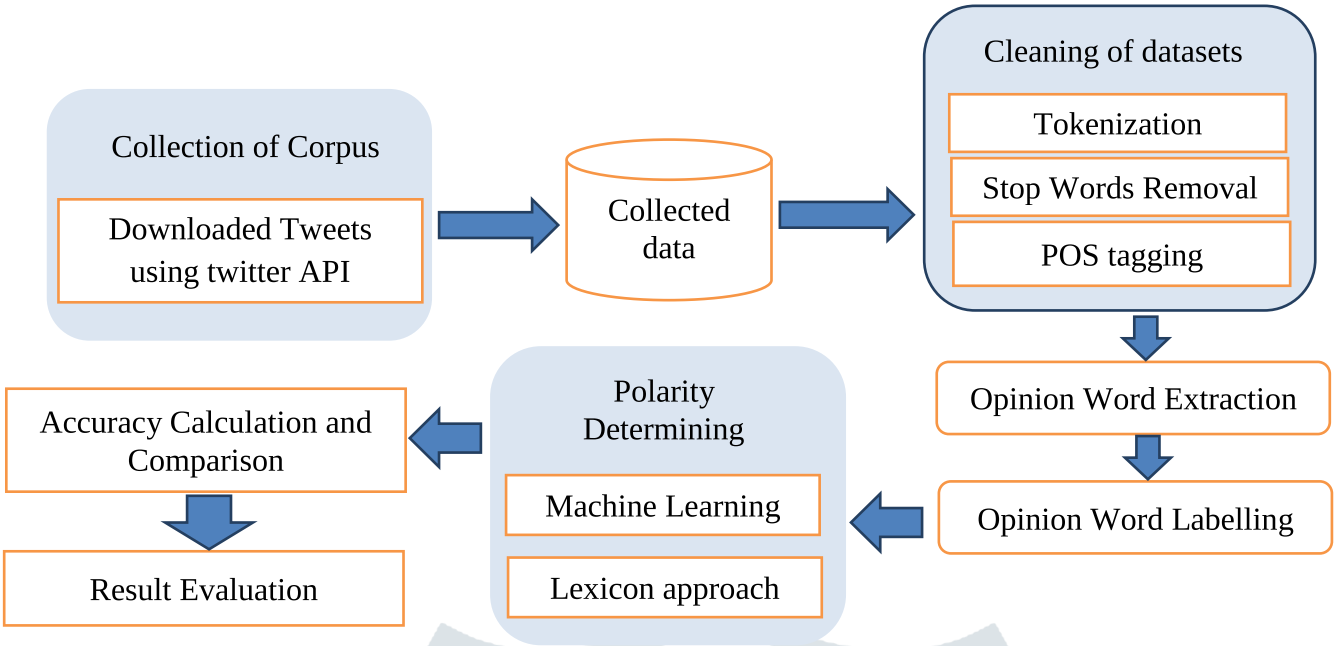

Figure 4: Twitter Sentiment Analysis Process

All steps shown in the above figure are explained as following:

a. Corpus collection: Collecting labelled datasets is one of the crucial challenges in Twitter's sentiment analysis. The Twitter API is used to collect a collection of text posts. Then, these tweet posts are combined together to construct a dataset of three classes: positive emotions, negative emotions, and a group of objective texts (no emotion).

b. Data Cleaning: Data received from Twitter typically consists of multiple HTML objects such as < > & which are embedded in the original data [12]. Therefore, it is essential remove these objects. The Twitter dataset also contains other information such as retweets, hashtags, usernames and modified tweets. All of these are overlooked and eliminated from the dataset. These main steps in data cleaning are:

- Tokenization: Tokens denote individual words or terms. The purpose of tokenization is to split a thread of text into tokens.
- Stop Word Removal: All human languages consist of plenty of Stop words. By removing these words, the low-level information is removed from the text to bring more attention to important information. Removing stop words certainly decreases the dataset dimensionality and thus lessens the training time owing the smaller number of tokens used in training.
- POS tagging: Parts of speech such as adjectives, adverbs and verbs and certain groups of nouns are decent pointers of subjectivity and emotion. Syntactic dependency patterns can be generated by parsing or dependency trees [13].

c. Opinion Word Extraction: The Twitter language model has several exclusive features. Some of these features are considered to decrease the feature space. First of all, to initiate process, all unigrams and bigrams in the corpus above a certain limit are extracted. For example, all unigrams and bigrams with frequencies greater than 5 are taken out as candidate features. In general, Unigrams and Bigrams are selected in word/phrase level sentiment analysis [14]. It can also be easily stretched using trigrams. Then for each tweet, the frequency of each candidate features discovered in it is calculated. Accordingly, the following feature vector is created by means of the term frequency for every tweet:

({word1:frequency1, word2:frequency2 …}, "polarity")

d. Opinion Word Labelling: The aggregated words are then verified on a dictionary of positive words and negative words containing two files, and a polarity is given to the tweet based on that. If the tweet contains any positive word or hash tag, it will be assigned as +1 and -1 will be assigned for each negative word.

e. Polarity Determining: In the last step, the polarity of a tweet is determined [15]. The impact of pre-processing on sentiment classification is estimated by employing any one of the two strategies, i.e., lexicon-based approach or machine learning.

2.1 Sentiment Analysis on Twitter data using Deep Learning

Vishu Tyagi, et.al (2020) suggested a CNN- LSTM (Convolutional Neural Network-Long Short Term Memory) based DL (deep learning) technique [21]. Moreover, pre-trained embedding approach was also adopted for extracting the attribute in automatic manner so as sentiments were analysed, and reviews gathered from Twitter were classified as positive or negative. A publicly available dataset was utilized to implement the suggested technique and obtaining the results. The results depicted that the suggested technique outperformed the baseline techniques. A Hadoop framework and DL (deep learning) classification algorithm was intended by Mudassir Khan, et.al (2020) to analyse the sentiment. The data was distributed using Hadoop cluster in order to extract the attributes [22]. Afterward, the twitter data had provided the considerable attributes. The DRNN (deep recurrent neural network) classification model was adopted for allocating areal-valued review to each input twitter data. The input data was classified as positive review and negative review using this model. The intended approach had provided the accuracy around 0.9302, sensitivity up to 0.9404 and higher specificity around 0.9157 in comparison with traditional approaches. A NN (neural network)-based technique was projected by Marco Pota, et.al (2018) with the objective of analysing the sentiment expressed on political tweets [23]. Initially, the dense vectors consisted of sub word information were utilized to depict the text. The morphology and semantics were employed for detecting word similarities. Subsequently, the way for classifying the tweets on the basis of sentiment was learned by training CNN (Convolutional Neural Network) on a labelled dataset. Finally, the projected technique was implemented for analysing the sentiments from collection of tweets extracted during the days ahead of the latest UK General Election. The outcomes revealed that the projected technique was more efficient in contrast to others while classifying the tweets as positive or negative. A word embedding technique generated from unsupervised ML (machine learning) was put forward by Zhao Jianqiang, et.al (2018) on huge twitter corpora [24]. The latent contextual semantic relationships and co-occurrence statistical attributes were considered among words in tweets in this approach. A sentiment feature set of tweets was generated by integrating these work embeddings with n-grams attributes and word sentiment polarity score. The DCNN (deep convolution neural network) algorithm made the deployment of this feature set in order to train and predict the sentiment classification labels. Five Twitter data sets were executed to compare the presented approach with traditional models in the experimentation. The results validated that the presented approach had provided superior accuracy and F1-measure while classifying the twitter sentiments.

Comparison Table 1

Author	Year	Technique used	Dataset	Results
Vishu Tyagi, et al.	2020	CNN- LSTM (Convolutional Neural Network-Long Short-Term Memory)	SentiSW	The suggested technique outperformed the baseline techniques.
Mudassir Khan, et al.	2020	Hadoop framework and deep learning classifier	Twitter API dataset	The intended approach had provided the accuracy around 0.9302, sensitivity up to 0.9404 and higher specificity around 0.9157.
Marco Pota, et al.	2018	A neural network-based approach	Twitter API	The projected technique was more efficient in contrast to others while classifying the tweets.
Zhao Jianqiang, et al.	2018	Word embedding method	Five twitter data sets	The presented approach had provided superior accuracy and F1-measure while classifying the twitter sentiments.

2.2 Sentiment Analysis on Twitter data using Machine Learning

Ruchi Mehra, et.al (2017) focused on analysing the sentiment behaviour extracted from the Twitter data [25]. Thus, a hybrid approach was investigated in which NB (Naive Bayes) was integrated with Fuzzy classification algorithm for analysing the twitter sentiments and classifying the tweets into three categories such as positive, negative or neural behaviour. The results obtained in analysis confirmed the superiority of intended approach over existing techniques during the classification procedure. The intended approach performed more effectively with regard to accuracy, precision and recall. An Apache Spark framework, an open source distributed data processing platform, was introduced by Hossam Elzayady, et.al (2018) for analysing the sentiments in which distributed memory abstraction was exploited [26]. The Apache Spark's MLIB (Machine learning library) aimed at handling an extraordinary volume of data in effective manner. Some stages to pre-process the data and to extract the text feature were presented to provide outcomes for analysing the sentiment. A comparative analysis was conducted for quantifying the introduced framework with regard to scalability. The results demonstrated that the introduced framework had generated more optimal results as compared to other techniques. A corpus-based approach was developed by Hussain Al Salman, et.al (2020) to analyse the Arabic sentiment of tweets and classify them as negative or positive in twitter social media [27]. This approach was planned on the basis of DMNB (Discriminative multinomial naïve Bayes) with N-grams tokenize and TF-IDF (term frequency-inverse document frequency). The publicly available twitter dataset was applied for testing the developed approach during the experimentation with regard to diverse metrics. The experimental outcomes indicated that the developed approach was efficient and performed better as compared to other technique. This approach enhanced the accuracy by 0.3%.An ensemble classification model was recommended by Ankit, et.al (2018) in which the base learning classifiers namely NB (Naive Bayes), RF (Random Forest), SVM (Support Vector Machine) and LR (Logistic Regression) were integrated for constructing the recommended model so that sentiments were analysed from Twitter [28].The outcomes indicated that the recommended model was more effectual in contrast to others. This approach assisted the companies in monitor consumer opinions related to their product and consumers in selecting the best products.

Comparison Table 2

Author	Year	Technique used	Dataset	Results
Ruchi Mehra, et al.	2017	A hybrid of naive Bayes and Fuzzy classifier	Twitter REST API	The intended approach performed more effectively with regard to accuracy, precision and recall.
Hossam Elzayady, et al.	2018	Apache Spark model	Twitter API	The results demonstrated that the introduced framework had generated more optimal results concerning scalability.
Hussain AlSalman, et al.	2020	Discriminative multinomial naïve Bayes (DMNB) and TF-IDF	RapidMiner tool	The developed approach was efficient and enhanced the accuracy by 0.3%.
Ankit, et al.	2018	Ensemble classifier	Twitter API dataset	The recommended model was more effectual in contrast to others.

2.3 Sentiment Analysis on Twitter data using Optimization Technique

T. Sreenivasulu, et.al (2021) established the fuzzy based IDNN (Intelligent deep neural network) along with the chaotic-PSO (particle swarm optimization) for analysing the twitter sentiment so that the service of airline industries was analysed [29]. The extraction of tweets was done. The considerable attributes were extracted by pre-processing these tweets and classified them in three classes: positive, negative or neutral. The integration of IDNN and chaotic PSO worked effectively. The results demonstrated that the established technique was capable of handling imbalance huge dataset in big data paradigm and attaining superior accuracy. Two new models known as SAW (Sentimental All-Weather) and SMPT (Sentimental MPT) were suggested by Edmund Kwong Wei Leow, et.al (2021) to gather current market conditions by Twitter sentiments extracted from Google's BERT (Bidirectional Transformer). These models were optimized using GA (Genetic Algorithm) to accomplish diverse tasks such as to increase the cumulative returns and mitigate the volatility [30]. The tweets and the United States stock data had considered for training these models. The results depicted that the suggested models performed well with regard to diverse metrics such as Sharpe ratio, cumulative returns, and value-at-risk in comparison with other methods while analysing the Twitter sentiments. A hybrid ML (machine learning) technique was constructed by Mohammad A. Hassonah, et.al (2019) for analysing the sentiments [31]. The SVM (Support Vector Machine) algorithm was implemented to construct a classifier on the basis of 3 classes: positive, neutral, and negative. For this, the ReliefF algorithm was integrated with MVO (Multi-Verse Optimizer). A dataset having 6900 tweets was extracted from Twitter for testing the constructed technique. A comparative analysis was conducted on constructed approach with respect to accuracy. The results revealed the superiority of the constructed approach over other techniques. A new GA (genetic algorithm) was intended by Hamidreza Keshavarz, et.al (2017) for dealing with optimization problem and discovering lexicons for analysing the sentiments [32]. Afterward, this algorithm was utilized to extract the meta-level attribute which was utilized with the Bing Liu's lexicon and n-gram features. Six datasets were executed to carry out the experiments. The results validated that the intended algorithm had yielded the accuracy around80% and F-measure 80% above. The sentiment lexicons generated from the intended algorithm assisted in analysing the culture of Twitter users and sentiment orientation of words in diverse contexts.

Comparison Table 3

Author	Year	Technique used	Dataset	Results
T. Sreenivasulu, et al.	2021	Intelligent Deep Neural Network integrated with Chaotic Particle Swarm Intelligence	Twitter API	The established technique was capable of handling imbalance huge dataset in big data paradigm and attaining superior accuracy.
Edmund Kwong Wei Leow, et al.	2021	Robo-advisor using genetic algorithm and BERT	Google's Bidirectional Transformer (BERT)	The suggested models performed well with regard to diverse metrics such as Sharpe ratio, cumulative returns, and value-at-risk.
Mohammad A. Hassonah, et al.	2019	Relief and Multi-Verse Optimizer (MVO)	Twitter API	The results revealed the superiority of the constructed approach over other techniques concerning accuracy.
Hamidreza Keshavarz, et al.	2017	Adaptive lexicon learning using genetic algorithm	six datasets	The intended algorithm had yielded the accuracy around80% and F-measure 80% above.

3. Research Methodology

In this work, sentiment analysis is performed on twitter data. The important steps followed in the novel methodology are mentioned below:

a) Extraction of Microblogs data and its pre-processing : Different clients post information in different forms in the form of tweets to express their sentiments on variety of topics. The pessimistic and affirmative are the two categorizations among which the Twitter data sample is applied. Tweeter data is generally collected using Twitter API. Twitter API stands for Application Programming Interface. Twitter API facilitates software developers to access and interrelate with openly available Twitter data. In order to interact with this API, Developers may write their own scripts or may use one of the public libraries accessible in various programming languages.

In general, two APIs are used by the Twitter API to retrieve tweets in significant manner. These are:

- **Twitter Streaming API:** This API enables the interaction of streaming Twitter data and collects tweets in realistic way. It is possible to listen in all the Tweets corresponding to a certain keyword, mention or hashtag, along with collection of tweets of particular customers while they are posting tweets on the Twitter platform.
- **Standard Search API:** This API provides past tweets posted up to 7 days ago, corresponding to a predefined query (the keyword, mention, hashtag, etc. that you'd like to search). Different from real-time analysis, the information of past can be retrieved using this API.

b) Pre-processing: After capturing tweets required for sentiment analysis, the next step is to prepare the data. The data on social media exist in raw form. It implies that this data is noisy, rough and required cleaning. This is a vital step as the quality of the data will bring about more consistent outcomes. There are several tasks involved in preprocessing a Twitter dataset. For example, eliminating all sorts of inappropriate information such as emojis, special characters, and additional blank spaces. It may also perform more tasks such as improving format; deleting duplicate tweets, or tweets smaller than three characters. The lexical analysis is done using n-gram algorithm which can process the input information.

c) Feature Extraction: There are several properties included in the preprocessed data sample. The features of developed data sample are extracted using the characteristic extraction method. Further, in a phrase, the optimistic and pessimistic polarity is calculated such that the individuals using replicas can be formatted. To perform dispensation, there are few machine learning methods that require representation of key features of contents. The characteristic vectors used for performing categorization are used for measuring input characteristics. This work makes use of N-grams for feature extraction. A brief description of this approach is provided below:

- **N-grams:** N-grams of texts are widely employed to perform several tasks related to text mining and NLP (Natural Language Processing). These are mainly a set of co-existing words inside a specified window. In order to compute the n-grams, the movement of one word is done in forward direction

If variable X represents number of words in a given sentence K, the number of n-grams for sentence K would be:

$$Ngrams_K = X - (N - 1)$$

There are several tasks which can be performed using N-grams. For example, in order to develop a language model, n-grams are employed for not only developing unigram models but also develop bigram and trigram models. Google and Microsoft have developed web scale n-gram models. These models can be employed to carry out several tasks. These tasks include spelling correction, word breaking and text summarization. The one more aim of using n-grams is to develop features for supervised Machine Learning models such as SVM, MaxEnt models, Naive Bayes, and so on. The plan is to make use of tokens e.g. bigrams in the feature space rather only unigrams.

d) Training: For providing solutions to categorization issues, managed learning is known to be an important technique. To perform prospect forecasting of unidentified information, it is easier to perform training of classifier. To extract the dataset features, KNN classifier method is applied. To define the centroid points, k-mean approach is applied by KNN classifier. From these points, the Euclidian distance is calculated. In one class, the similar points are categorized. K-Nearest Neighbor is a very machine learning algorithm. This algorithm depends on supervised learning approach. This approach makes assumption about the similarity amid the novel case/data and accessible cases. This approach place the novel case into the category most analogous to the existing categories. This algorithm stores all the existing data and performs the classification of a new data point on the basis of similarity. It implies that new data can be effortlessly classified into an appropriate category with the help of this approach. This algorithm can be used for both Regression as well as Classification. However, it is mainly employed for the classification issues. It is a non-parametric algorithm. It means that this algorithm does not assume any underlying data. It is also known as a lazy learner algorithm. This algorithm does not learn from the training set instantaneously rather than its stores the dataset. During classification, this algorithm works on the dataset. At the training stage, this approach merely stores the dataset. After getting novel data, this algorithm performs the classification of this data into a category much analogous to the novel data.

4. Result and Discussion

Twitter is extensively utilized social media platform. This platform has numerous personal profile pages of millions of users. The personal data of users is comprised in this page. Users are allowed to follow each other for communication purpose and obtaining the access of content of other users in easy manner. As per the survey, the generation of 500 million tweets done in every day. The posts on tweeter contain millions of opinions. Thus, twitter has attained a lot of attention in the research field for opinion mining as it is incorporated with a variety of web-based applications.

The earlier work has various classification models for predicting the accuracy of ML (machine-learning) algorithms. Precision is a part of relevant extracted examples. In case of class, the precision is division of number of accurate results such as TPs (true positives) and number of all returned results such as the total of TPs and FPs (false positives) in classification.

Table 4: SVM Classification Performance Analysis

	SVM			
	PRECISION	**RECALL**	**F1-SCORE**	**ACCURACY**
Dataset1	0.79	0.80	0.80	80.23
Dataset2	0.86	0.86	0.85	85.57
Dataset3	0.82	0.82	0.82	81.76
Dataset4	0.88	0.88	0.85	87.51
Dataset5	0.87	0.87	0.87	86.82
IMDB Dataset.csv	0.89	0.89	0.89	88.73
dataset_en.csv	0.76	0.77	0.75	77.28
movie_reviews.csv	0.89	0.89	0.89	88.98
olympic2021_live.csv	0.89	0.87	0.84	87.32

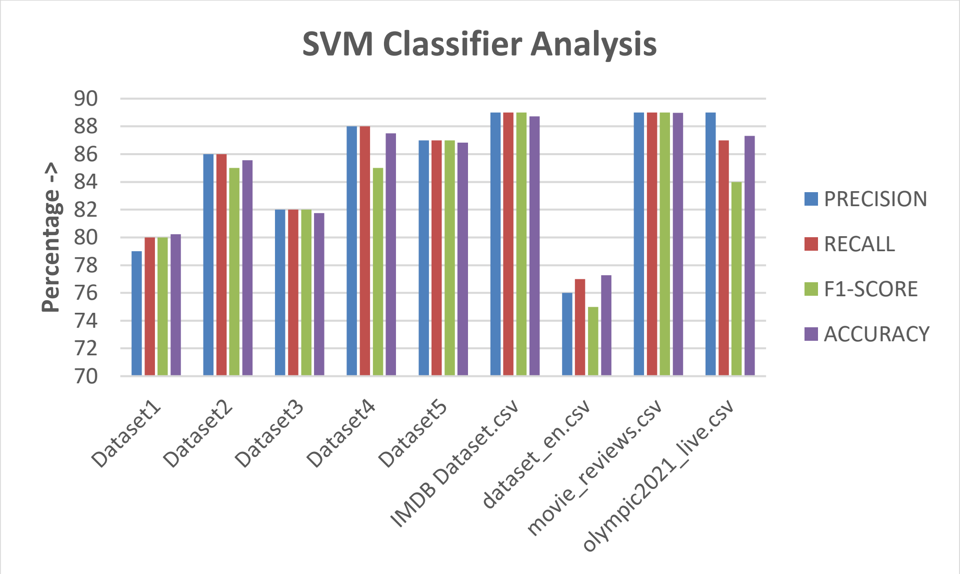

Figure 5: SVM Classifier Performance

As shown in figure 5, the SVM classifier is applied for the sentiment analysis of twitter data. The performance of the model is analysed in terms of precision, recall, F1-measure and accuracy. The performance of the SVM classifier is tested in the different datasets.

Table 5: SVM+KNN Classifier Performance Analysis

	SVM+KNN			
	PRECISION	**RECALL**	**F1-SCORE**	**ACCURACY**
Dataset1	0.70	0.68	0.67	78.48
Dataset2	0.84	0.84	0.84	96.35
Dataset3	0.72	0.70	0.69	80.38
Dataset4	0.82	0.84	0.81	96.62
Dataset5	0.78	0.78	0.77	89.64
IMDB Dataset.csv	0.77	0.75	0.74	85.76

dataset_en.csv	0.68	0.70	0.67	80.58
movie_reviews.csv	0.77	0.76	0.75	87.09
olympic2021_live.csv	0.83	0.85	0.82	90.79

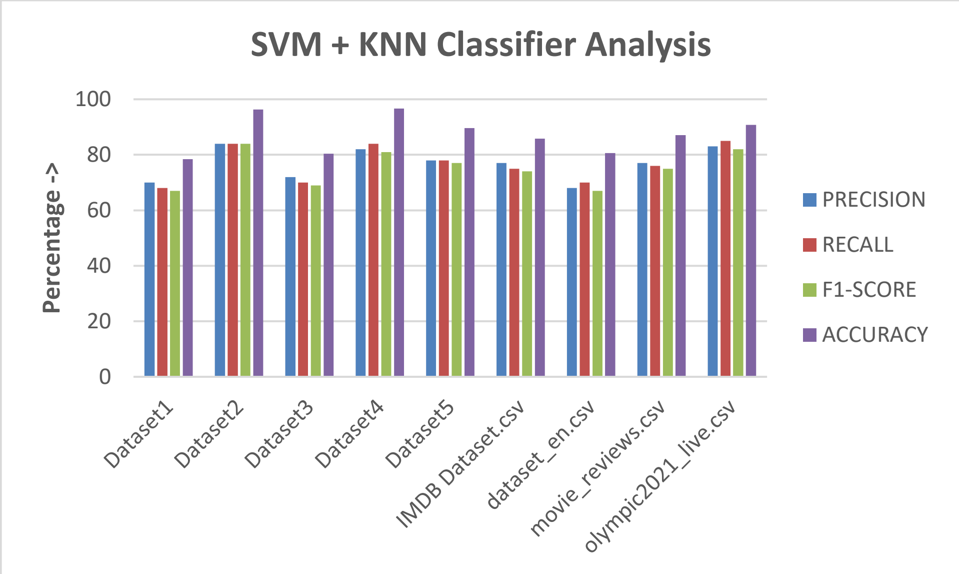

Figure 6: SVM+KNN Classifier Performance

As shown in figure 6, the SVM+KNN classifier is applied for the sentiment analysis of twitter data. The performance of the model is analysed in terms of precision, recall, F1-measure and accuracy. The performance of the SVM+KNN classifier is tested in the different datasets.

Table 6: Performance Analysis of Voting Classifier

	VOTING CLASSIFIER(Logistic Regression, Random Forest Classifier +SVM+KNN)			
	PRECISION	**RECALL**	**F1-SCORE**	**ACCURACY**
Dataset1	0.82	0.82	0.82	90.73
Dataset2	0.86	0.85	0.85	93.72
Dataset3	0.83	0.83	0.83	91.48
Dataset4	0.86	0.85	0.86	93.50
Dataset5	0.85	0.85	0.85	93.19
IMDB Dataset.csv	0.89	0.89	0.89	92.43
dataset_en.csv	0.78	0.78	0.78	81.20
movie_reviews .csv	0.88	0.88	0.88	91.94
olympic2021_l ive.csv	0.87	0.86	0.87	94.92

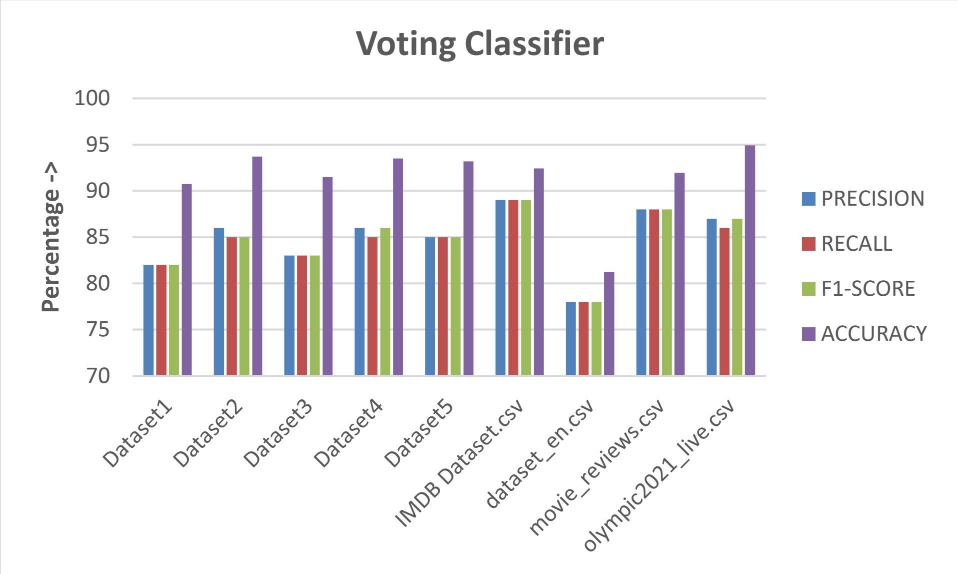

Figure 7: Voting Classifier Performance

As shown in figure 7, the voting classifier is applied for the sentiment analysis of twitter data. The performance of the model is analysed in terms of precision, recall, F1-measure and accuracy. The performance of the voting classifier is tested in the different datasets.

Table 6: Time and Space Complexity Analysis

DATASET	COMPLEXITY		SVM	SVM+KNN	VOTING CLASSIFIER
Dataset1	Time		4.03 s ± 257 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	1.36 s ± 125 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	5.9 s ± 178 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	50	50	51.3
		CPU	8	8	8
Dataset2	Time		2.58 s ± 115 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	1.9 s ± 93.5 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	3.9 s ± 248 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	47.2	47.2	47.3
		CPU	8	8	8
Dataset3	Time		3.84 s ± 129 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	1.41 s ± 80.8 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	5.33 s ± 163 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	47.8	48.1	48
		CPU	8	8	8
Dataset4	Time		3.35 s ± 200 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	1.6 s ± 141 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	5.65 s ± 247 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	48.3	48.4	47.9
		CPU	8	8	8

Dataset5	Time		6.13 s ± 374 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	2.21 s ± 118 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	9.5 s ± 433 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	48.3	48.2	48.6
		CPU	8	8	8
IMDB Dataset.csv	Time		9min 22s ± 1min 32s per loop (mean ± std. dev. of 7 runs, 1 loop each)	1min 39s ± 8.2 s per loop (mean ± std. dev. of 7 runs, 1 loop each)	10min 20s ± 1min 48s per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	45.2	55.1	50
		CPU	8	8	8
dataset_en.csv	Time		4min 3s ± 6.9 s per loop (mean ± std. dev. of 7 runs, 1 loop each)	34.4 s ± 1.37 s per loop (mean ± std. dev. of 7 runs, 1 loop each)	7min 47s ± 20.4 s per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	44	54.3	44
		CPU	8	8	8
movie_reviews.csv	Time		12min 3s ± 1min 72s per loop (mean ± std. dev. of 7 runs, 1 loop each)	1min 54s ± 1.25 s per loop (mean ± std. dev. of 7 runs, 1 loop each)	13min 22s ± 1min 92s per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	46	46.3	48.8
		CPU	8	8	8
olympic2021_live.csv	Time		1.45 s ± 44.5 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	1.01 s ± 86.1 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)	4.13 s ± 47.9 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
	Space	RAM	49.4	48.9	60.8
		CPU	8	8	8

Conclusion

The feature extraction method established relationship between attribute and target set. In the last step of classification, the classification method is enforced which can categorize data into certain classes like positive, negative and neutral. In the previous method, the hybrid classification method is applied to evaluate the sentiments of the twitter data, but still, there's some room to improve accuracy and precision. In this study, a hybrid classification method is designed which is the mixture of Gaussian Naive Bayes and random forest classifier and SVM Classifier for the sentiment analysis. The performance of hybrid classifier is analyzed in terms of accuracy, precision and recall. The proposed model is implemented on four datasets and maximum accuracy is achieved up to 93.72 percent for the sentiment analysis. It is examined that outcome for the sentiment analysis of the proposed model is optimized up to 4 to 5 percent approximately. In future the propose technique can be further extended using transform learning.

References

[1] Srivastava, Akansha, and Ravindra Gupta. "Sentiment Analysis Techniques: A Review."

[2] Eirinaki, Magdalini, Shamita Pisal, and Japinder Singh. "Feature-based opinion mining and ranking." Journal of Computer and System Sciences 78.4 (2012): 1175-1184.

[3] Haddi, Emma, Xiaohui Liu, and Yong Shi. "The role of text pre-processing in sentiment analysis." Procedia Computer Science 17 (2013): 26-32.

[4] Ghag, Kranti, and Ketan Shah. "Comparative analysis of the techniques for sentiment analysis." 2013 International Conference on Advances in Technology and Engineering (ICATE). IEEE, 2013.

[5] Vuong, Quang Hong, and Atsuhiro Takasu. "Transfer learning for emotional polarity classification." 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). Vol. 2. IEEE, 2014.

[6] Shahana, P. H., and Bini Omman. "Evaluation of features on sentimental analysis." Procedia Computer Science 46 (2015): 1585-1592.

[7] Tripathy, Abinash, Ankit Agrawal, and Santanu Kumar Rath. "Classification of Sentimental Reviews Using Machine Learning Techniques." Procedia Computer Science 57 (2015): 821-829.

[8] Farooq, Umar, et al. "A word sense disambiguation method for feature level sentiment analysis." 2015 9th international conference on software, knowledge, information management and applications (SKIMA). IEEE, 2015.

[9] Ahmed, Shoiab, and Ajit Danti. "Effective sentimental analysis and opinion mining of web reviews using rule based classifiers." Computational Intelligence in Data Mining—Volume 1. Springer, New Delhi, 2016. 171-179.

[10] Bouazizi, Mondher, and Tomoaki Otsuki Ohtsuki. "A pattern-based approach for sarcasm detection on twitter." IEEE Access 4 (2016): 5477-5488.

[11] Tsytsarau, Mikalai, and Themis Palpanas. "Managing diverse sentiments at large scale." IEEE Transactions on Knowledge and Data Engineering 28.11 (2016): 3028-3040.

[12] Saleena, Nabizath. "An ensemble classification system for twitter sentiment analysis." Procedia computer science 132 (2018): 937-946.

[13] Naz, Sheeba, Aditi Sharan, and Nidhi Malik. "Sentiment classification on twitter data using support vector machine." 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI). IEEE, 2018.

[14] Hiremath, Savitha, S. H. Manjula, and K. R. Venugopal. "Unsupervised Sentiment Classification of Twitter Data using Emoticons." 2021 International Conference on Emerging Smart Computing and Informatics (ESCI). IEEE, 2021.

[15] Phand, Shital Anil, and Jeevan Anil Phand. "Twitter sentiment classification using stanford NLP." 2017 1st international conference on intelligent systems and information management (ICISIM). IEEE, 2017.

[16] Rane, Ankita, and Anand Kumar. "Sentiment classification system of twitter data for US airline service analysis." 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). Vol. 1. IEEE, 2018.

[17] Ibrahim, Mohd Naim Mohd, and Mohd Zaliman Mohd Yusoff. "Twitter sentiment classification using Naive Bayes based on trainer perception." 2015 IEEE Conference on e-Learning, e-Management and e-Services (IC3e). IEEE, 2015.

[18] Sharma, Parul, and Teng-Sheng Moh. "Prediction of Indian election using sentiment analysis on Hindi Twitter." 2016 IEEE international conference on big data (big data). IEEE, 2016.

[19] Barnaghi, Peiman, Parsa Ghaffari, and John G. Breslin. "Opinion mining and sentiment polarity on twitter and correlation between events and sentiment." 2016 IEEE second international conference on big data computing service and applications (BigDataService). IEEE, 2016.

[20] Minab, Shokoufeh Salem, Mehrdad Jalali, and Mohammad Hossein Moattar. "A new sentiment classification method based on hybrid classification in twitter." 2015 international congress on technology, communication and knowledge (ICTCK). IEEE, 2015.

[21] Tyagi, Vishu, Ashwini Kumar, and Sanjoy Das. "Sentiment Analysis on Twitter Data Using Deep Learning approach." 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN). IEEE, 2020.

[22] Khan, Mudassir, and Aadarsh Malviya. "Big data approach for sentiment analysis of twitter data using Hadoop framework and deep learning." 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE). IEEE, 2020.

[23] Pota, Marco, et al. "A subword-based deep learning approach for sentiment analysis of political tweets." 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA). IEEE, 2018.

[24] Jianqiang, Zhao, Gui Xiaolin, and Zhang Xuejun. "Deep convolution neural networks for twitter sentiment analysis." IEEE Access 6 (2018): 23253-23260.

[25] Mehra, Ruchi, et al. "Sentimental analysis using fuzzy and naive bayes." 2017 International Conference on Computing Methodologies and Communication (ICCMC). IEEE, 2017.

[26] Elzayady, Hossam, Khaled M. Badran, and Gouda I. Salama. "Sentiment Analysis on Twitter Data using Apache Spark Framework." 2018 13th International Conference on Computer Engineering and Systems (ICCES). IEEE, 2018.

[27] AlSalman, Hussain. "An improved approach for sentiment analysis of arabic tweets in twitter social media." 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS). IEEE, 2020.

[28] Saleena, Nabizath. "An ensemble classification system for twitter sentiment analysis." Procedia computer science 132 (2018): 937-946.

[29] Sreenivasulu, T., and R. Jayakarthik. "Intelligent Deep Neural Network integrated with Chaotic Particle Swarm Intelligence based Sentiment Analysis in Big Data Paradigm." 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS). Vol. 1. IEEE, 2021.

[30] Leow, Edmund Kwong Wei, Binh P. Nguyen, and Matthew Chin Heng Chua. "Robo-advisor using genetic algorithm and BERT sentiments from tweets for hybrid portfolio optimisation." Expert Systems with Applications 179 (2021): 115060.

[31] Hassonah, Mohammad A., et al. "An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter." Knowledge-Based Systems 192 (2020): 105353.

[32] Keshavarz, Hamidreza, and Mohammad Saniee Abadeh. "ALGA: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs." Knowledge-Based Systems 122 (2017): 1-16.