



## Employee Sentiment Analysis Towards Remote Work during COVID-19 Using Twitter Data

Nagaratna Parameshwar Hegde<sup>1\*</sup>      Sireesha Vikkurty<sup>1</sup>      Gnyanee Kandukuri<sup>1</sup>  
 Sriya Musunuru<sup>1</sup>      Ganapatikrishna Parameshwar Hegde<sup>2</sup>

<sup>1</sup> Department of Computer Science and Engineering,  
 Vasavi College of Engineering (Autonomous) Hyderabad, India

<sup>2</sup> Department of Information Science and Engineering, SDM Institute of Technology, Ujire, India

\* Corresponding author's Email: nagaratnaph@staff.vce.ac.in

---

**Abstract:** The COVID-19 pandemic has essentially transformed the way of leading a life for millions of people across the world. As offices remained closed for months, employees expressed conflicting sentimental analysis on the work-from-home culture. People worldwide use social media platforms such as Twitter to talk about their daily lives made a trend in the online platform. This research study aims to gauge the public's sentiment on working from home/ remote locations during the COVID-19 pandemic by tracking their opinions on Twitter. The existing random forest model trained the data faster but failed to predict the results faster. Therefore, an ensemble model is proposed to predict an outcome using a distinct modeling algorithm. An ensemble classifier has been used for enhancing the performances using the base learning classifiers such as Naive Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR) form an Ensemble classifier. The proposed ensemble model aggregates each base model for the prediction and results for the unseen data. These tokens are then passed to the Ensemble classifier that classifies the sentiments and assigns a polarity (positive, negative, neutral) to every tweet. The proposed Ensemble method improve the average prediction performance over any contributing member in the ensemble. The results obtained by the proposed Ensemble model reached accuracy of 97.47 % when compared to the existing models such as Deep LSTM, SVM model that obtained accuracy of 83 %, 84.46 %.

**Keywords:** COVID-19 pandemic, Ensemble model, Sentiment analysis, Twitter, Work-from-home.

---

### 1. Introduction

Tweets posted in the live Twitter timeline are used to generate the set of data and accessed through tweepy application programming interface (API). About 2 lakh tweets are relevant to the remote work during the pandemic performed tokenization using the TextBlob library. This study aims to gauge the public's sentiment on working from home/remote locations during the COVID-19 pandemic by tracking their opinions on Twitter [1]. It is essential to study these trends at this point in the pandemic as organizations should decide whether to continue remote work indefinitely or reopen offices and workspaces, depending on productivity and employee satisfaction. The world has seen an

increase in the usage of internet websites which increased the use of social media and microblogging platforms such as Twitter, Facebook, and Instagram for API in recent years [2]. The social platforms have enormous amounts of data coming directly from the users. Twitter, a microblogging site, generates approximately 500 million tweets a day, with users sending out about 6000 tweets every second. Analyzing this barrage of data existing on the internet can be used to gauge public sentiment [3-5]. On social networks/media sites, users discuss topics ranging from politics to movie reviews, conversations on social issues, and brand marketing. These discussions reflect the mindset of the users and can be used to gauge the public sentiment as a whole if analyzed systematically [6]. The data collected from these sites can be beneficial for companies and

media organizations. This data can be used for organizations to analyze what people think about their products and services. Generally, the polarity of these statements is categorized as “positive”, “negative” or “neutral” [7]. As sentiment analysis deals with the natural language used by users, it is essential to examine the statement to understand its polarity closely [8-10]. Sentiment analysis employs three sub-tasks to employ this process and identify the overall polarity accurately:

1. Determination of a phrase to be positive, negative, or neutral is done using Phrase-level determination.
2. Sentences in a given message are marked positive, negative, or neutral. For sentences that express mixed sentiments, whichever polarity is stronger is chosen.
3. Document analysis emphasizes aligning the whole message in the document as positive, negative, or neutral.

Therefore, in the present research, an ensemble model is proposed that helps in improving the results obtained using combined multiple models known as ensemble models. The ensemble model will allow for better predictions when compared with the single model. The proposed method combines, naive bayes, RF, SVM, LR to enhance the performances in terms of accuracy using base learning techniques. The results obtained from the proposed ensemble classifier considered the majority voting to classify the data and obtained better sentiments such as positive, negative, and neutral.

The organization of the paper is as given as follows: section 2 describes the existing methods involved in sentiment analysis using Twitter data. section 3 describes the proposed ensemble classifier. section 4 describes the results and discussion that includes quantitative and comparative analysis. section 5 discusses the conclusion and future work.

## 2. Literature review

The existing models that were undergone for sentimental analysis based on the tweets are as follows:

Imran [11] developed a cross-cultural polarity for emotion detection using COVID-19 related tweets to perform sentiment analysis under deep learning techniques. The main objective of the research was to overcome the anxiety and thus considered the reaction of citizens’ sentiment about the actions executed by distinct countries for overcoming the pandemic. In the developed model, the deep long

short term memory (LSTM) model was considered for the estimation of the emotions and sentiment polarity from the tweets. The word embedding models like GloVe and BERT captured a similar context. The developed model showed a limitation in understanding the sarcastic context and it failed to understand the same word usage and was not able to discriminate the contexts.

Kaur [12] performed sentiment analysis using deep learning algorithms to analyze COVID-19 tweets. The developed model analyzed the Twitter data based on the hashtag words, keywords, related to COVID-19, coronavirus, deaths, etc., The developed model designed a hybrid heterogeneous support vector machine (H-SVM) performed sentiment classification and classified into positive, neutral, and negative scores. However, the model showed limitations when built on different language tweets, Facebook and Instagram data.

Chakraborty [13] performed sentiment analysis using the evolutionary based LSTM model for classification of COVID-19 tweets. The developed model collected large number of tweets for sentiment analysis on COVID-19 tweets and analyzed the public sentiment on the topics for evolutionary classification for n-gram analysis. The LSTM model used 2 types of rated tweets for the classification on COVID-19 data. However, the model failed to find the mostly recurred words for analyzing the polarity and it required more data to train the deep learning model for enhancing the validation accuracy.

Jianqiang [14] developed a depth convolution neural network (DCNN) to classify the sentiment of user on Twitter tweets. A sentiment feature set was formed by combining the word embedding with n-gram features and sentiment polarity score features. A DCNN model obtained these features set as an input for identifying the sentiment classification labels. The data sparseness problem was reduced by catching the context sentiment information and retain the word order information. The experiments were conducted on five datasets using important parameters, namely f-measure, precision-recall and accuracy. The performance of DCNN was affected because the parameter values were too large and unlabeled micro-blog messages were used to extract the inputs.

Wang [15] predicted the polarities of sentiments by implementing the SentiDiff that considered the inter-relationships between textual information and sentiment diffusion patterns of Twitter messages. A sentiment reversal was investigated to analyze the sentiment diffusion of tweets. The method collected the tweets and retweets from Beijing intelligent starshine information technology corporation, China.

The efficiency of SentiDiff was tested by using various parameters such as classification accuracy and area under the precision-recall curve (PR-AUC). The combination of textual and sentiment diffusion information had a negative influence on SA tasks that lead to poor performance on the sentiment reversal prediction model.

### 3. Proposed scheme

The block diagram of the proposed method is as shown in Fig. 1. The proposed scheme analyzes the opinions about work from home policies during the COVID-19 period and attempts to understand the sentimentality behind the users on a large scale. The high-level system design of the proposed structure is given in Fig. 1.

This structure is used to quantify opinion estimation of remote work policies. The suggested framework mainly consists of data extraction, pre-processing, and sentiment determination modules. These modules are discussed in detail in the below sections.

#### 3.1 Data extraction

This study aims to test user opinion as it changes in real-time. So, tweets that match the subject of the study are extracted from the live timeline. Twitter API is used to collect tweet data and stored it in a database. Tweepy is a Python library that is used to access the Twitter API. While searching for tweets, query matching with the tweets includes the geospatial information, user, etc., got down the exact data required to perform the evaluation. Internally, the Twitter API uses two streaming endpoints to deliver the tweets. The sampled stream endpoint is

used when the general mood of the audience and global events are to be monitored as it returns a sample of 1 % of the tweets appearing on the live timeline. For more specific queries like searching for tweets related to work from home policies during the COVID-19 period, the filtered stream endpoint can be used to apply filters and rules to track and listen to particular topics or events and get matching public tweets. Any one of these filtering criteria or combinations of these criteria can be specified for implementation. Acquiring data in parts at various points in time is done to increase the relevance of the data, and it is discouraged to collect data in a single iteration

Acquiring the data with the above-mentioned method leads to find the information which may or may not be helpful, and when the data collected is not applicable cleans the required data. Following is the list of key-value pairs similar to Python “dictionary” data type:

- favorite tweet or not
- User ID
- Users Screen name
- Original tweet or not
- Hashtags exist or not
- retweet or not
- Twitter user - registered account language
- Tweet Geo-tag location
- Timestamp of tweet creation

In the next section, the steps to clean this data and make it usable for analysis are highlighted.

#### 3.2 Data pre-processing

Tweets are highly non-structured, contain grammatical errors and non-grammatical data, linguistic variations, and extensive usage of acronyms and emoticons. So, it is crucial to process this data to perform any statistical analysis on it. For data pre-processing, the TextBlob library is used. It is a Python library that is used for text-processing operations through an easy-to-use API. TextBlob objects are internally treated as Python strings, and standard natural language processing (NLP) tasks are applied. The following series of data formatting techniques are applied to the raw data:

- Tokenization: A token is a sequence of characters that are grouped as a meaningful unit like symbols and words for further processing. The process of obtaining the tokens from the tweet text is called tokenization.

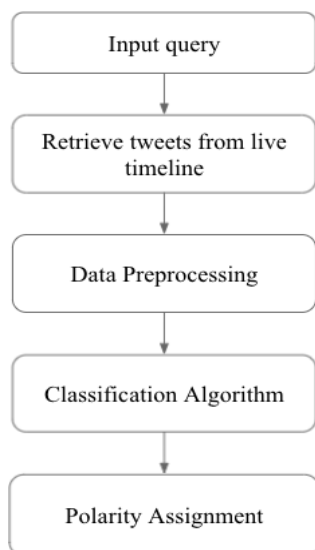


Figure. 1 Block diagram of the proposed research

Tokens are stored internally in a database.

```
>>> sen = TextBlob("Vasavi College of Engineering ")
>>> sen.words
WordList(['Vasavi', 'College', 'of', 'Engineering'])
>>> sen.sentences
[Sentence("Vasavi College of Engineering ")]
```

- **Cleaning:** User references (identified by the token "@") and URLs (identified by tokens "HTTP") are removed from the twee text.
- **Stemming:** It is the process of the normalizing text so that a derived word is reduced to its stem. For example, the root word “swim” is obtained by stemming the phrases “swimmer”, “swam” and “swimming”. To reduce the complexity in word comparison, ignoring the complex grammatical transformations of the word in the original text stemming is used.
- **Stop-words removal:** A few words such as “a,” “an,” “the,” “he,” “she,” “by,” “on,” etc., are used purely for grammatical and referential purposes which serve no purpose while analyzing the data. These words, called stop words, can be removed as they are ubiquitous, but give no additional information regarding the subject.
- **Parts-of-speech (PoS) tagging:** Based on the context of the sentence, each word in the sentence previously broken down into tokens is assigned a tag corresponding to the grammatical part of speech (noun, verb, adjective, adverb, etc.)

```
>>> from textblob import Word
>>> w = Word("ran")
>>> w.lemmatize()
'run'
```

```
>>> sen = TextBlob("JAVA is a high-level programming language.")
>>> sen.tags
[(('JAVA', 'NNP'), ('is', 'VBZ'), ('a', 'DT'), ('high-level', 'JJ'), ('programming', 'NN'), ('language', 'NN'))]
```

The processed data is suitable to apply statistical methods to determine the sentiment of a particular tweet, thereby determining the sentiment of the collected data as a whole. This process can be divided into two subtasks: Classification and polarity assignment.

### 3.3 Classification

The text in the tweets is now passed through a computing model to assign categories to the particular tweet. Classification is the process wherein the categorization of data is based on ways that they

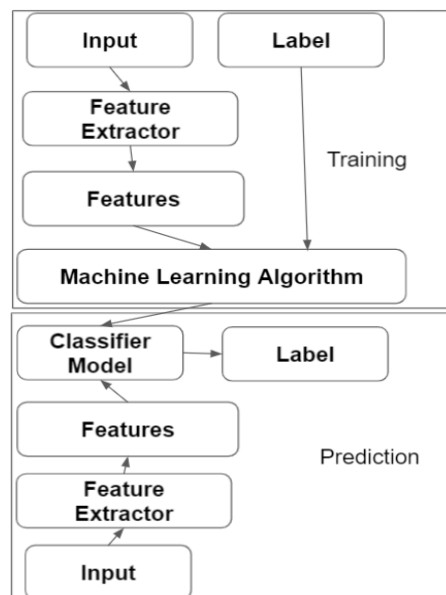


Figure. 2 Training and prediction modules

are alike i.e. Similar patterns exist in one class objects and differ from the patterns in the other class. In the proposed work, a classifier is designed so that it exactly categorizes tweets into four sentiment categories namely neutral, positive, negative, and ambiguous.

As demonstrated in the figure above, the system consists of two modules: Training and prediction. A pre-classified dataset is used to train the quantitative model so that the model can assess and learn the features that are categorized as a particular class as shown in Fig. 2. The processed Twitter data is then passed to the trained model for feature extraction. The features are passed to a classifier model that is used to assign polarity based on the underlying mathematical machine learning algorithm. The classifier model is chosen in such a way that it gives maximum accuracy to the type of data that is chosen. The naive bayes classifier is used in this study.

#### 3.3.1. Naive bayes

The naive bayes algorithm works by predicting the probability of membership of a particular word for each class. The highest probability class is assigned as the most likely class. The following equation characterizes the mathematical reasoning behind the calculation of the unigram word models using naive bayes. Furthermore, if identical class sizes are assumed for objective ‘obj’ and subjective classes ‘subj’, the prior probability ‘P’ of the objective class can be ignored. Henceforth, the following equation can be formulated, where  $\prod$  is the product of ‘ $w_i$ ’ that stands for the  $i^{th}$  word ranging from 1 to  $N \in$  Whole number, is shown in Eq. (1).

$$P(obj|tweet) = \frac{\prod_{i=1}^N P(w_i|obj)}{\prod_{i=1}^N P(w_i|obj) + \prod_{i=1}^N P(w_i|subj)} \quad (1)$$

The class of a word is predicted based on the highest probability using a maximum a posteriori (MAP) method. The MAP is used for defining the following formula.

$$\begin{aligned} MAP(H) &= \max(P(H|E)) \\ &= \max(P(E|H) * P(H)) / P(E) \\ &= \max(P(E|H) * P(H)) \end{aligned} \quad (2)$$

$P(H)$  is known as the probability of hypothesis  
 $P(E)$  is evidence probability which is calculated using Eq. (2).

### 3.3.2. Random forest(RF)

RF is an ensemble classifier that consists of a decision tree classifier that builds and sets up itself for training the dataset. Once after the collections of votes from the decision trees will decide the final class or label for testing the object. The RF classifier consist of single votes from 1 to  $B \in$  Whole numbers for assigning the frequent class  $C_b$  to input vector  $x$  that is carried out by using Eq. (3).

$$C_r^B = Majorityvote\{C_b(x)\}_1^B \quad (3)$$

### 3.3.3. Support vector machine (SVM)

The support vector machine (SVM) model is used for training the data and is a probabilistic classifier. It uses a nonlinear mapping to find the large margin among the classes. The training time required by SVM was slowed due to high accuracy. The SVM finds the decision boundary that maximizes the separation gap among the classes and results with good accuracy for overcoming the Twitter SA problem.

### 3.3.4. Logistic regression(LR)

The proposed regression model is used for the classification of categorical dependent variables from the independent variables. The LR will attempt for finding the hyperplane to extend the separation gap maximally among the classes.

### 3.3.5. Proposed ensemble classifier

An ensemble classifier will aggregate the base classifiers to obtain the robust classifier for enhancing the performances. An ensemble classifier has been used for enhancing the performances using the base learning classifiers such as NB, RF, SVM, LR form ensemble classifier.

The working of the ensemble classifier in the proposed method is as follows:

### 3.3.6. Ensemble model calculates sentiment score of the tweet based on the majority vote

Step 1: The proposed system model is trained with the training data

Step 2: The set of tweets are used for testing the system. Each of the base classifiers such as NB, SVM, DT, LR, and RF are used for determining the sentiment as (positive, negative, neutral) of each of the test tweets.

Step 3: For each of the base classifier the classification report was calculated for testing the test tweet.

Step 4: The probability for each of the tweets is calculated into classifying them as negative, positive, or neutral.

Step 5: The probability is assigned for each weight of the classifier under the ensemble technique using each of the classifiers.

Step 6: The algorithm will calculate the positive and negative scores on the tweet based on the prediction obtained by each of the classifiers.

### 3.3.7. Polarity assignment

In the final module of the system, the emphasis is on the determination of sentiment. SA is used for text categorization. The sentiment classification is performed for contextual sentiment analysis categorized the specific tweets based on the given context. The contextual sentiment analysis is performed for the categorization of significant parts of tweets with respect to the given context. The sentiment analysis will deal with the entire text the sentiment. The proposed schema follows general sentiment analysis as the study's goal is to determine user sentiment towards remote work during the COVID-19 pandemic. So, it is vital to understand the sentiment as expressed in the tweet as a whole. The classifier used in this case, the NB classifier where the document 'd' is used from all out the classes where  $c \in C$ , the classifier gives the maximal posterior probability for the document. The maximum posterior probability 'argmax P' gives the document  $d$  as shown in the following Eq. (4) the appropriate class  $c$ .

$$c = \operatorname{argmax} P(c|d) \quad (4)$$

where  $c \in C$

The TextBlob package for Python is used to perform the natural language processing (NLP) tasks.

Sentiment analysis is performed on the parsed tweets by using the sentiment.polarity method of the TextBlob library. The tweets are assigned a polarity score in the range [-1.0, 1.0].

*sentiment.polarity* > 0 => 'positive'  
*sentiment.polarity* == 0 => 'neutral'  
*sentiment.polarity* < 0 => 'negative'

Along with a polarity score, the model also assigns a subjectivity score in the range [0.0, 1.0]. Here 0.0 is very objective and 1.0 is very subjective. Along with polarity and subjectivity, an intensity score is assigned to every word in the lexicon to determine how much it modifies the next word.

Let us consider the word 'very'. It has a polarity score of 0.3, subjectivity score of 0.3, and intensity score of 1.3. The negation of this word by adding 'not' before it multiplies the polarity by -0.5 but keeps the subjectivity constant. While POS tagging, it is identified that 'very' is a modifier. So in the phrase 'very great', TextBlob ignores polarity and subjectivity but takes intensity score into account while calculating the sentiment of the next word, 'great'. In this way, TextBlob assigns polarity and subjectivity to words and phrases and averages the scores for longer sentences/text.

According to sentiments expressed in the tweets, the tweets are labeled in three classes: positive, negative, and neutral.

- Positive: Tweet is considered as positive if the user expressed tweet has an upbeat/thrilled/ecstatic/cheery outlook or if the words mentioned have positive annotations.
- Negative: A tweet is considered as negative if the tweet has negative/angered/sad feelings or if words with negative annotations are mentioned.
- Neutral: Tweet is considered neutral if the user does not express his view/opinion in the tweet and merely communicates facts.

#### 4. Results

As the study's goal is to understand employees' opinions towards work from home policies enforced during the COVID-19 pandemic posted on Twitter, tweets generated during that period from all over the world were collected. The volume of tweets corresponding to the terms 'work-from-home', 'remote work', 'virtual work', and 'home offices' increased during this time. The results of the proposed Bi-GRU model with gaussian noise layer are simulated by anaconda navigator and Python 3.6

software with Windows 10 operating system. The proposed ensemble model was implemented in a system consisting of 128 GB RAM, 1 TB memory, configured with RTX 2080 Ti GPU, and i9 processor operating at 3 GHz.

#### 4.1 Performance measures

The proposed method uses the following parameters namely f-measure, accuracy, precision, and recall.

Precision measure the truly positive tweets, where a portion of positive tweets are determined by the recall for a given class  $c$  which is explained in Eq. (5 and 6).

$$Precision = \frac{|Tweets\ correctly\ classified\ to\ class\ c|}{|Total\ tweets\ in\ class\ c|} \quad (5)$$

$$Recall = \frac{|Tweets\ correctly\ classified\ to\ the\ class|}{|Total\ tweets\ in\ class\ c|} \quad (6)$$

To determine the sentimental analysis measurement of statistical variability and random errors are used to define the overall accuracy, which is explained in Eq. (7).

$$Accuracy = \frac{|Total\ number\ of\ correctly\ classified\ tweets|}{|Total\ number\ of\ tweets|} \quad (7)$$

The mathematical Eq. (8) provides the f-score calculation by considering the recall and precision of the test are measured as.

$$F - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (8)$$

#### 4.2 Quantitative analysis

The analysis found that the sentiment expressed in the first six months after the remote work period was majorly positive. While this sentiment took a more neutral route during the next six months, eventually, after a year of working from home, the users expressed a negative outlook towards the culture. The general volume of tweets also increased during this time, peaking as per significant world events and the result obtained by VADER considered 300 tweets for the evaluation as shown in Table 1. The performance of the proposed work based on naïve bayes classification is done by finding accuracy and precision using a confusion matrix. VADER (valence aware dictionary and sentiment reasoner) is known as the rule-based sentiment analysis which is a lexicon specifically attuned for expressing the sentiments in the media [16]. Fig. 3 Comparison of

classification results and Table 2 shows the confusion matrix for the proposed ensemble method. similarly, Table 3 shows the confusion matrix for the sentiment analysis by VADER.

The accuracy of the proposed multi-classification is given by the average accuracy of the 3 classes so accuracy of all 3 classes is listed as in Table 4 and the comparison of accuracy term is shown graphically in Fig. 4.

The testing of the model reveals that the bayesian classification method is better than VADER in finding the sentiments of the employees is denoted as in Table 5. Table 6 shows the results obtained for the proposed method in terms of accuracy, precision, recall, and f-measure and Fig. 5 represents the performance measure obtained for the proposed model.

The proposed ensemble model considered 2 lakhs of tweets for the evaluation of results. Table 7 shows the results obtained by applying the proposed bayesian method.

Table 1. Classification details

	Positive Tweets	Negative Tweets	Neutral Tweets
Manual classification of tweets	92	35	173
Tweets classified by VADER	126	33	141
Tweets classified by the proposed method	80	36	184

Table 2. Confusion matrix for the proposed method

Total tweets =300		True Tweet class		
		Positive	Negative	Neutral
Predicted tweet class	Positive	71	5	4
	Negative	7	24	5
	Neutral	14	6	164

Table 4. Accuracy comparison

Class	Accuracy by Naïve Bayes (proposed)	Accuracy by VADER
Positive Tweet	0.91	0.70
Negative Tweet	0.92	0.87
Neutral Tweet	0.90	0.76

Table 6. The results obtained for the proposed method in terms of accuracy, precision, recall, and f-measure

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F- Score (%)
Naive Bayes	73.65	77.96	77.18	77.57
Random Forest	70.61	68.73	92.13	78.73
Support Vector Machine	74.36	76.06	82.56	79.18
Logistic Regression	73.44	73.89	85.08	79.09
Proposed ensemble model	97.47	96.62	92.18	99.30

Fig. 6 shows that the overall attitude towards work from home has been positive with 28 % of users showing a favorable sentiment towards it. While

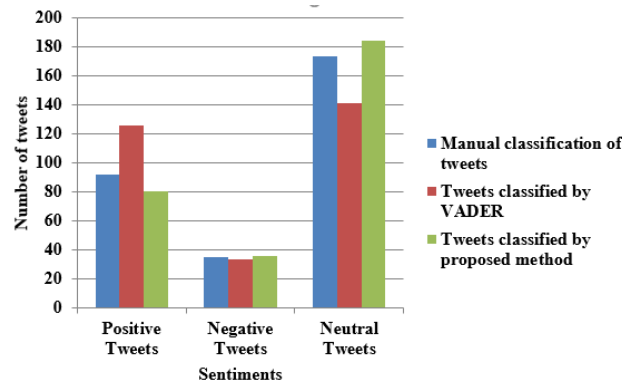


Figure. 3 Comparison of classification results.

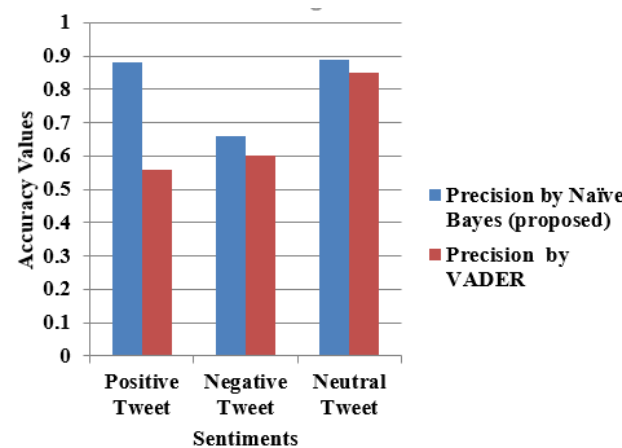


Figure. 4 Comparison of accuracy of classification

Table 3. Confusion matrix for the sentiment analysis by VADER

Total tweets =300		True Tweet class		
		Positive	Negative	Neutral
Predicted tweet class	Positive	65	10	41
	Negative	12	20	1
	Neutral	15	5	121

Table 5. Precision comparison

Class	Precision by Naïve Bayes (proposed)	Precision by VADER
Positive Tweet	0.88	0.56
Negative Tweet	0.66	0.60
Neutral Tweet	0.89	0.85



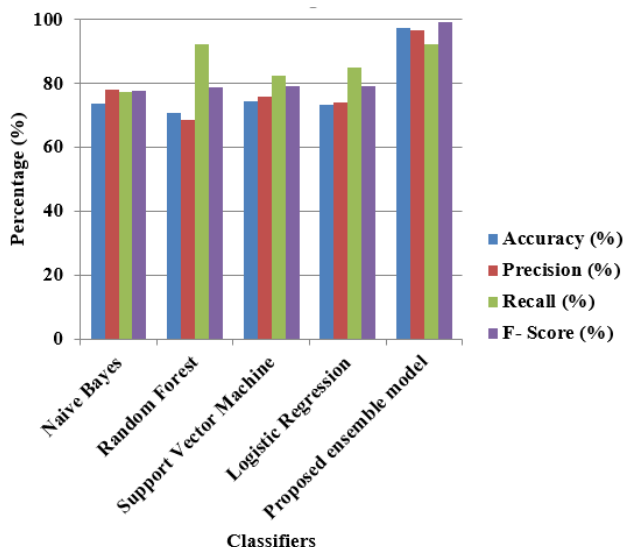


Figure. 5 The performance measure obtained for the proposed model

Table 7. classification of the tweets

Total tweets considered	Positive Tweets	Negative Tweets	Neutral Tweets
200000	56400	26600	117000
100%	28.20%	13.30%	58.50%

**Employee Sentiment analysis Towards Remote Work During COVID-19**

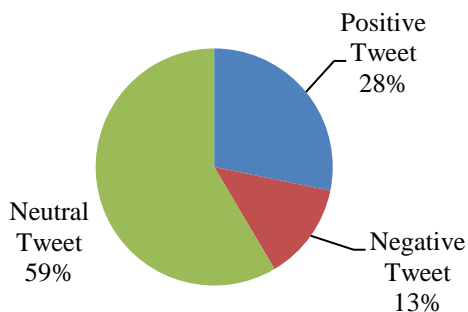


Figure. 6 Employee sentiment towards remote work during COVID-19

13 % of users expressed a negative sentiment, 59% of users had a neutral attitude.

**4.3 Comparative analysis**

Table 8 shows the comparative analysis for the proposed and the existing model in terms of accuracy, precision, and f-score obtained with respect to the Twitter dataset.

Table 8 shows the values obtained by the existing methodologies for sentiment analysis for tweets classification based on the impact of COVID 19 on workers. The MKH-SVM [12] obtained an accuracy

Table 8. Comparative analysis

Authors	Method	Accuracy (%)	Precision (%)	F1 score (%)
Imran [11]	Deep LSTM model	83	-	78
Kaur [12]	MKH-SVM	96.3	86	77
Chakraborty [13]	Sequential LSTM	84.46	81	81
Jianqiang [14]	Deep CNN	85.63	87.60	87.50
L. Wang [15]	Textual information and Sentiment Diffusion Patterns	-	-	80.91
Proposed method	Ensemble model	97.47	96.62	99.30

of 96.3 %, where the number of each data point are used as features exceeds the training data samples underperform using SVM. Similarly, accuracy of 83 % was obtained by the existing deep LSTM model [11], [13] for tweet classification. LSTMs faces over fitting problems and will be difficult in applying the dropout algorithm for overcoming the issues. The DCNN [14] model obtained the features that identified the sentiment classification labels but they were affected because of the too large parameter values and unlabeled micro-blog messages for extraction of inputs resulted with accuracy of 85.63 %. Whereas, in [15] combined the textual and sentiment diffusion information showed a negative influence for SA lead for poor performances in terms of f-score of 80.91 % for sentimental prediction. The regularization model with a dropout has been used for inputting the recurrent and input connects to the LSTM units excluded the weight and activation updates when the network has been trained. When compared with the existing methods, the proposed ensemble model that involved multiple classifiers such as NB, RF, SVM, and LR obtained 97.47 % of accuracy. The proposed ensemble method improved the average prediction performance over any contributing member in the ensemble. The mechanism for improved performance with ensembles is often the reduction prediction errors made by the contributing models. The results of the comparative analysis showed that the existing deep



LSTM model showed accuracy of 83 %, MKH-SVM of 96.3 %, sequential LSTM of 84.46 %, deep CNN of 85.63 %. Whereas, the proposed ensemble model obtained accuracy of 97.47 % that showed improvement in accuracy.

## 5. Conclusion

The research findings emphasize the positive effects of the pandemic on the overall population and improved the results from the work from home policy. In light of the changing effect of working in isolation on employees, organizations might consider shifting to workspaces as the sentiments slowly turned more negative to evolve the study period. Thus, considering the opinions of Twitter users, these findings have important implications for organizations to maintain their productivity levels. The proposed ensemble model will improve the ML results when multiple models were combined to produce better predictions. The classifiers such as NB, RF, SVM, LR are combined to enhance the performances in terms of accuracy for base learning models. The proposed ensemble model showed better accuracy performances by majority voting ensemble obtained better when compared to other classifiers. The results obtained by the proposed ensemble model gained accuracy of 97.47 % when compared to the existing models. The existing models such as deep LSTM, and SVM model obtained accuracy of 83 %, 84.46 % lower when compared with the proposed ensemble learning model. However, in the future, the computational complexity created in the system has to be reduced to improve the performance.

## Conflicts of interest

The authors declare no conflict of interest.

## Author contributions

The paper background work, conceptualization, methodology, dataset collection, implementation, result analysis and comparison, preparing have been done by 3<sup>rd</sup> and 4<sup>th</sup> author. The supervision, review of work and project administration, has been done by 1<sup>st</sup> and 2<sup>nd</sup> author. Editing draft and visualization have been done by 5<sup>th</sup> author.

## Acknowledgments

We thank Vasavi College of Engineering (Autonomous), Hyderabad for the support extended towards this work.

## References

- [1] Z. Long, R. Alharthi, and A. E. Saddik, "Needfull—A tweet analysis platform to study human needs during the COVID-19 pandemic in New York state", *IEEE Access*, Vol. 8, pp. 136046-136055, 2020.
- [2] T. Pano and R. Kashef, "A Complete VADER-Based Sentiment Analysis of Bitcoin (BTC) Tweets during the Era of COVID-19", *Big Data and Cognitive Computing*, Vol. 4, No. 4, p. 33, 2020.
- [3] Z. Yao, J. Yang, J. Liu, M. Keith, and C. Guan, "Comparing tweet sentiments in megacities using machine learning techniques: In the midst of COVID-19", *Cities*, Vol. 116, p. 103273, 2021.
- [4] O. Gencoglu and M. Gruber, "Causal Modeling of Twitter Activity During COVID-19", *arXiv preprint arXiv:2005.07952*, 2020.
- [5] H. S. Kisan, A. P. Suresh, and H. A. Kisan, "Collective intelligence & sentimental analysis of twitter data by using Stanford NLP libraries with software as a service", In: *Proc. of International Conf. on Computational Intelligence and Computing Research*. pp. 1-4, 2016.
- [6] P. H. Shahana and B. Omman, "Evaluation of Features on Sentimental Analysis", *Procedia Computer Science*, Vol. 46, pp. 1585-1592, 2015.
- [7] H. Rosenberg, S. Syed, and S. Rezaie, "The Twitter pandemic: The critical role of Twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic", *Canadian Journal of Emergency Medicine*, Vol. 22, No. 4, pp. 418-21, 2020.
- [8] S. Padmavathia and E. Ramanujam, "Naïve Bayes Classifier for ECG abnormalities using Multivariate Maximal Time Series Motif", *Procedia Computer Science*, Vol. 1, No. 47, pp. 222-8, 2015.
- [9] S. A. A. Subaihini and S. H. A. Khalifa, "A system for sentiment analysis of colloquial Arabic using human computation", *The Scientific World Journal*, 2014.
- [10] Z. Jianqiang and G. Xiaolin, "Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis", *IEEE Access*, Vol. 5, pp. 2870-2879, 2017.
- [11] A. S. Imran, S. M. Doudpota, Z. Kastrati, and R. Bhatra, "Cross-cultural polarity and emotion detection using sentiment analysis and deep learning--a case study on COVID-19", *arXiv preprint arXiv:2008.10031*, 2020.
- [12] H. Kaur, S. U. Ahsaan, B. Alankar, and V. Chang, "A Proposed Sentiment Analysis Deep Learning

- Algorithm for Analyzing COVID-19 Tweets”, *Information Systems Frontiers*, pp. 1-13, 2021.
- [13] A. K. Chakraborty, S. Das, and A. K. Kolya, “Sentiment Analysis of Covid-19 Tweets Using Evolutionary Classification-Based LSTM Model”, In: *Proc. of Research and Applications in Artificial Intelligence*, Springer, Singapore, pp. 75-86, 2021.
- [14] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, “Deep convolution neural networks for twitter sentiment analysis”, *IEEE Access*, Vol. 6, pp. 23253-23260, 2018.
- [15] L. Wang, J. Niu, and S. Yu, “SentiDiff: Combining Textual Information and Sentiment Diffusion Patterns for Twitter Sentiment Analysis”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 32, No. 10, pp. 2026-39. 2019.
- [16] C. J. Hutto and E. E. Gilbert, “VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text”, In: *Proc. Of Eighth International Conference on Weblogs and Social Media (ICWSM-14)*, Vol. 8, No. 1, pp. 216-225, 2014.