Machine Learning based Sentiment Analysis towards Indian Ministry

K. Bhargavi[1,*], Pratish Mashankar[2], Pamidimukkala Vasista Sreevarsh[3], Radhika Bilolikar[4], Preethi Ranganathan[5]

[1]Assistant Professor, [2,3,4,5]Student
Department of Information Technology
Keshav Memorial Institute of Technology affiliated with Jawaharlal Nehru Technological University
Narayanguda, Hyderabad 500029

[1]bhargavikumbham@kmit.in, pratishsmashankar@gmail.com,[2]sreevarshvasista@gmail.com,
[4]rbilolikar232@gmail.com,[5]preethiranganathan17899@gmail.com

Abstract: This paper presents the performance of Twitter Sentiment Analysis on the Agricultural Ministry of India and also determines the optimal AI model between Random Forest (RF) and k Nearest Neighbors (kNN) Algorithms. 'Twitter and Reddit Sentimental Analysis' dataset from Kaggle is used to train and test the RF and kNN AI models. Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer is used to extract features from the obtained dataset. The optimal AI model is determined using the library functions of scikit-learn-2.0 which perform the training and testing of the dataset. Twitter authenticated API called Tweepy is used to create the dataset required to determine the public sentiment towards the Agriculture Ministry by us. This dataset includes 6000 tweets related to the Agriculture Ministry. The Twitter Sentiment Analysis performed determines what percentage of public opinion towards the Agriculture Ministry is positive, negative and neutral.

Keywords: Twitter Sentiment Analysis, Random Forest Algorithm, K Nearest Neighbours Algorithm, Tweepy, Agriculture, Tweets.

1. Introduction

Social media has been through a major expansion especially during the COVID-19 pandemic resulting in it being the foremost utility of the Internet undertaken by many people around the globe. Twitter is one such platform which has 192 million energetic users and around 500 million tweets dispatched out in step with day regarding various topics from all over the world. A recent survey has found that there are around 22 million Indian users on Twitter which also includes accounts of government officials and ministries. With the ever-increasing number of 'Netizens', a term defined for users of the Internet, and their subsequent involvement in politics through public opinion, there is a big number of statistics recorded in the form of tweets directed towards diverse ministries and governments. Twitter has also emerged as a major platform in India owing to its popularity, ease of use and the sense of community that it fosters. Thus, it is clear that the corpus obtained from Twitter which is under consideration is humongous and can provide satisfactory results for any demographic analysis. In this paper, we performed Sentiment Analysis to classify tweets into any one of three categories, namely negative neutral and positive. This process of analysing and classifying tweets based on their polarity is defined as Twitter Sentiment Analysis, which is likely known as opinion mining or emotion AI. In order to study this, we began with the gathering of the related tweets from the Twitter database. We considered tweets to be related to a particular Ministry if they contained keywords referencing said Ministry. In this paper, the accuracy rates of the Random Forest Algorithm and the k Nearest Neighbors algorithm were compared and the algorithm with the higher accuracy rate was employed for performing the Twitter Sentiment Analysis on user microblogs related to the Indian Ministry of Agriculture.

2. Related Work

The Sentiment Analysis is to check the opinion of people on the schemes by the Central Government in recent years with the help of Twitter Data Analysis. These chosen schemes' tweets are classified based on the polarity and later are classified based on the opinions as positive, negative and neutral [1]. In this research, Twitter Sentiment Analysis (TSA) is performed to determine which party is performing better in the 2019 Indian General Elections by classifying user microblogs as positive, negative or neutral using three algorithms particularly, Support Vector Machine (SVM), Naïve Bayes Classifier and k-Nearest Neighbor (kNN). The kNN algorithm proved to be more efficient than the other two [2]. Pankaj Verma et al. proposed a twitter sentiment evaluation approach using R-Programming to apprehend public sentiment towards Indian government projects [3]. They

used the 'Syuzhet' R package for the classification of the microblogs. However this package does not include popular negative words like 'not' [4] which might produce a higher error rate. This paper produced the Support Vector Machines (SVM) and Recurrent Neural Networks-Long Term Short Memory (RNN- LSTM) algorithms to perform twitter sentiment analysis for determining public opinion towards government policies [5]. In particular, this paper intends to propose sentiment analysis techniques of a lexicon-based approach, machine learning-based approach, and hybrid approach for Twitter used by governments with their citizens [6]. This paper explained the study by conducting a twitter sentiment analysis for the Indian Union Budget 2020 with +149.3387 positive score using the Python TextBlob library, a rule based approach [7]. As feature based approaches provide greater accuracy than rule based approach [8] we are proposing machine learning techniques in our paper.

This study proposed TSA based on social media Twitter datasets of particular schemes and the polarity of sentiments using R libraries [9]. This paper used n-gram and parts-of-speech tags as features to train a Na¨ıve Bayes classifier and to determine a linguistic analysis of the collected corpus to determine positive, negative and neutral sentiments for a document [10]. Sentiment analysis of the Twitter dataset that expresses an opinion about Prime Minister of India's Digital India Campaign is performed where the tweet is gathered and categorized into positive, negative, or neutral [11]. This paper proposed TSA to determine the public opinion towards Dell Laptop using Naive Bayes algorithm with eight different features [12].

This study proposed a Twitter Sentiment Analysis while comparing three machine learning algorithms namely Naive Bayes, Logistic regression and SVM, and concluded that SVM outperforms the other algorithms [13]. Similarly, this paper compared a kNN inspired model with SVM and concluded that SVM underperformed while performing sentiment analysis on twitter data [14]. This paper compared Random Forest (RF) Algorithm with SVM while performing a sentiment analysis of the social media network and concluded that RF has better accuracy than SVM [15]. We thus employed a comparison between RF and kNN to determine the better algorithm between the two.

3. Methodology

The steps followed to determine the public sentiment towards the Indian Agriculture Ministry using the optimal algorithm between Random Forest Algorithm and k Nearest Algorithm is illustrated in Figure 1. The first step was to collect the corpus required for the study from Twitter and prepare the dataset needed. After this, we cleaned the dataset accordingly. We performed the Feature Vectorization after which the optimal classifier was trained to conclude the task of assigning a positive or a negative sentiment to an entire tweet. We then determined the accuracy of this prediction and presented the result in a visual representation. The steps are explained in the sections below

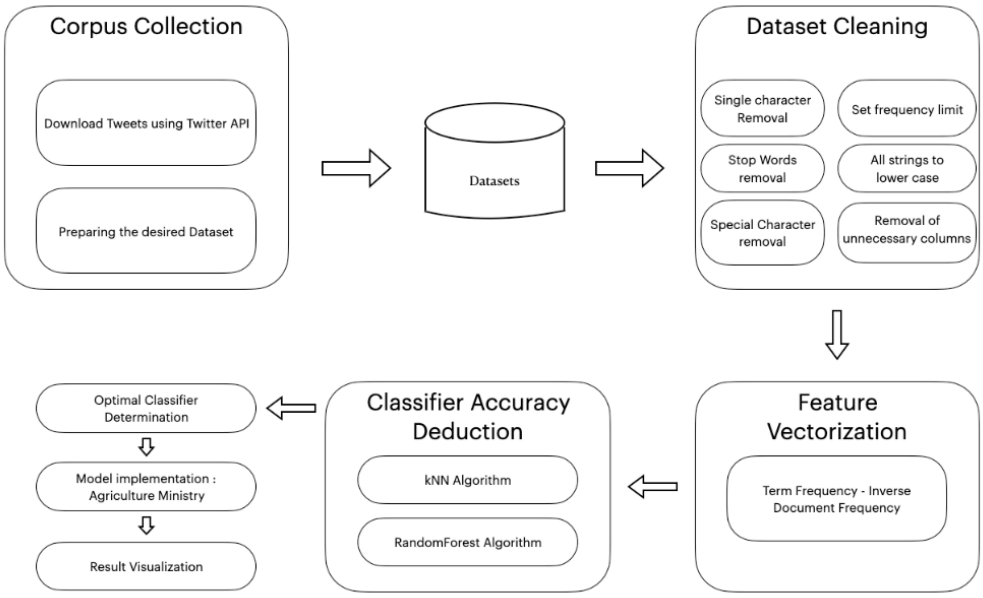

Fig. 1. Methodology of Twitter Sentiment Analysis to understand the public opinion towards Indian Agriculture Ministry using KNN and Random Forest algorithms

3.1 Corpus collection and dataset description

Twitter API is used in order to access the tweets posted by all the users on twitter. The API permits us to get right of entry to tweets after specifying the access token key, access secret key, consumer key and consumer secret that are specific to each task. After the authentication of the unique keys comes creating an API reference in the project. The next step is to retrieve the tweets which are needed for the project. To specify the search criteria for the tweets, we use the Cursor Object to get the reference of the location of the tweets fulfilling the criteria. This criterion is set by specifying the parameters for search. Some of the search parameters and values in this paper include language as 'en', geolocation-based searching by using geotag starting from the geographical center of the country with a radius of 1608km, and having the tweet_mode as 'extended' as to get the entire tweet. To specifically gather these tweets, we had used search_keywords to look for these keywords in the tweet and if present to fetch that tweet. Various keywords such as #Farmbills2020, which was one of the top trending hashtags at that moment on Twitter, @nstomar which is the official Twitter handle of the Agricultural Minister, Shri Narendra Tomar, and keywords such as 'agriculture' and 'farmers' had been used to gather the DataFrames required for the Agricultural dataset. All these individual DataFrames were then appended into one final dataset, which was converted to a CSV file. This dataset constructed by us consisted of 12000 tweets and four rows namely text, created_at, source and retweets which gave the utf-8 text of the tweet, UTC time at which the tweet was created, application used to post the tweet, and number of times the tweet was retweeted, that is, shared by other users on the microblogging platform respectively.

3.2 Dataset Cleaning

The dataset now contained numerous unnecessary columns which were beyond the scope of our methodology and hence were dropped. Furthermore, the general natural language processing data cleaning methods were employed to improve the accuracy of the model. This was achieved using regular expressions. These methods included removal of special characters, single characters, replacing multiple spaces with single space, and converting the string into lowercase. Further methods like frequency limit and stop words removal were performed during the feature vectorization. These methods were also employed on the labelled dataset required to train the classifier models. The description of this labelled dataset is given in section 3.4 of this paper

3.3 Feature Vectorization

We used Term Frequency and Inverse Document Frequency (TF-IDF) Vectorizer in order to provide better computational accuracy to the model. For any document (tweet), TF-IDF Vectorizer is used in order to replace the text with its numerical value i.e. with the weight of each word contained, by first considering a set of max_features. This set of max_features can be varied to improve the accuracy of a model. The min_df and max_df attributes of the TF-IDF function set lower and upper limits respectively of a particular word's frequency. Only the words' whose frequency is in between the upper and lower limits are termed as useful. The vectorizer calculates the weights for only the useful words and the weights for the rest is set to zero. TF-IDF vectorizer also removes all the stop words which do not add much meaning to the entire text. The output of a TF-IDF vectorizer i.e. the vectorizer object is then transformed into an array. This array is used to train a supervised machine learning model. The formula is given as:

$$tfidf(w,d,D) = tf(w,d) * idf(w,D) \quad (1)$$

$$tf(w,d) = \log(1 + f(w,d)) \quad (2)$$

$$idf(w,D) = \log(Nf(w,D)) \quad (3)$$

Where *tfidf(w,d,D)*is used to classify documents, ranking them in search engines. *tf(w, d)* is the term frequency i.e. the count of the words present in a document from its own vocabulary. *idf(w, D)* is the inverse document frequency i.e. importance of the word to each document. For a dataset, *N* represents the number of documents, *d* represents a particular document, *D* represents the collection of all documents, while *w* represents a particular word in that document. *f(w, d)*specify the frequency of word *w* in document *d.*We observed that a net of 3,038 features is generated upon the vectorization of tweets obtained from Twitter. And nearly 16,000 features are generated from the vectorization of labelled dataset. From these we selected the top 250, 500 and 750 features to compare the two classification algorithms.

3.4 Classifier accuracy deduction

Supervised Machine Learning (ML) is the principle of training a model using labelled data to predict an output. We determined which among the two popular supervised ML algorithms, namely the k-Nearest Neighbours and Random Forest, gives the best accuracy to perform sentiment analysis. The labelled data is obtained from Kaggle's Twitter and Reddit Sentimental analysis Dataset [16]. It consists of 1,62,980 tweets labelled as 1 (positive), 0 (neutral) and -1 (negative). This labelled dataset, after undergoing data cleaning and vectorization using TF-IDF, results in a vectorizer object. It is inferred from 3.3 that the vectorizer object generated from the labelled dataset is stored in an array variable named as processed_features. This contains the vectorized document (tweet) and the labelled sentiment. To determine the accuracy of any given model, it is required to split the array into testing and training data. We then obtain two sets of vectorized documents and labels. Thus, the processed_features array is split into 4 parts considering the train_test_split function. These split parts, named X_train, X_test, y_train and y_test, indicate the tweet or the text (X) and the polarity value of the text (y) for the training and testing data. The ratio of splitting of the array is specified by the text_size attribute, which is a decimal number ranging from 0 to 1.0 specifying the percentage of training data and testing data from the gathered data. We use the value 0.2 which specifies 20% of the entire data is split so that 20% of it is considered as the Testing Data and 80% of the entire data is considered as the Training Data. With the abetment of this 80% Training Data, we train the model in order to help predict the values for the testing data. Once the model has been trained using X_train and y_train, the 20% testing data (X_test) is passed to the model. The classifier then predicts values for each document present in the X_test array by using these algorithms. These predicted values are stored in a variable called y_pred. The y_test and y_pred values are tallied against each other to determine how many X_test values were predicted accurately. This gives an indication of the accuracy of the model. Hence upon splitting the dataset we obtained 1,30,384 samples for training the data and 32,596 samples for testing the data. To ensure that the correct model between the kNN and RF is chosen, we train both the classifiers using multiple max_features during the TF-IDF vectorization. Here we have used 250, 500 and 750 features. The parameters used to compare these two classifiers are given in the classification report. We also determine the accuracy scores of each classifier against the chosen number of features.

Classification report: The quality of predictions for a classifier is displayed in the classification report. It has the following main components

Precision: Precision is described because the ratio of proper positives to the sum of true as well as false positives and this is the total expected positive values, given through the formula:

$$precision = \frac{TP}{TP + FP} \quad (4)$$

Recall: Recall/sensitivity is described because the ratio of true positives to the sum of true positives and false negatives, this is, the whole effective values within the observations, given by using the system:

$$recall = \frac{TP}{TP + FN} \quad (5)$$

F1-score: It's miles the weighted common of Recall and precision taking each false positives and false negatives under consideration. this is given via the components:

$$F1 - score = \frac{2 * recall * precision}{recall + precision} \quad (6)$$

Support: In the given dataset, the number of actual occurrences of the class is make out as support.

Here TP, TN, FP, FN are true positive, true negative, false positive and false negative correspondingly.

Accuracy score: The ratio of correctly expected observations i.e., the sum of true positives and true negatives to the total observations. This is given through the components:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (7)$$

After performing data cleaning and vectorization of labelled data, we employed our classifiers- kNN and RF. The confusion matrices, the classification reports and the accuracy scores of the kNN and RF algorithm for 250, 500 and 750 features are shown in the following Table 1.

Table 1. Precision, Recall, F1-Score and Accuracy of kNN and RF algorithms using 250, 500, 750, 3038 features

S. No	Method	Number of features	Polarity	Precision	Recall	F1-Score	Accuracy (%)
1	KNN	250	-1	0.49	0.31	0.38	54.75
			0	0.47	0.81	0.59	
			1	0.74	0.47	0.57	
2	KNN	500	-1	0.6	0.24	0.35	52.26
			0	0.43	0.92	0.59	
			1	0.82	0.36	0.5	
3	KNN	750	-1	0.65	0.22	0.32	49.13
			0	0.41	0.94	0.57	
			1	0.84	0.29	0.43	
4	RF	250	-1	0.64	0.35	0.46	65.64
			0	0.59	0.8	0.68	
			1	0.73	0.69	0.71	
5	RF	500	-1	0.76	0.47	58	74.41
			0	0.66	0.92	0.77	
			1	0.85	0.74	0.79	
6	RF	750	-1	0.81	0.53	0.64	77.61
			0	0.69	0.95	0.8	
			1	0.87	0.76	0.81	
7	RF	3038	-1	0.85	0.71	0.78	86.86
			0	0.85	0.97	0.9	
			1	0.9	0.87	0.88	

3.5 Optimal classifier determination

The results for determining the optimal classifier determination are tabulated in Table 1. We noticed that the Random Forest (RF) algorithm accomplished more than the k Nearest Neighbors (kNN) algorithm for 250, 500 and 750 features. The Random Forest classifier is hence trained on the previously utilized labelled dataset. This time, however, instead of 250, 500 or 750 features, a total of 3,038 features were used to train the model. We observed that the accuracy now spiked from 77.61% to 86.86%. This classifier was used to perform the sentiment analysis.

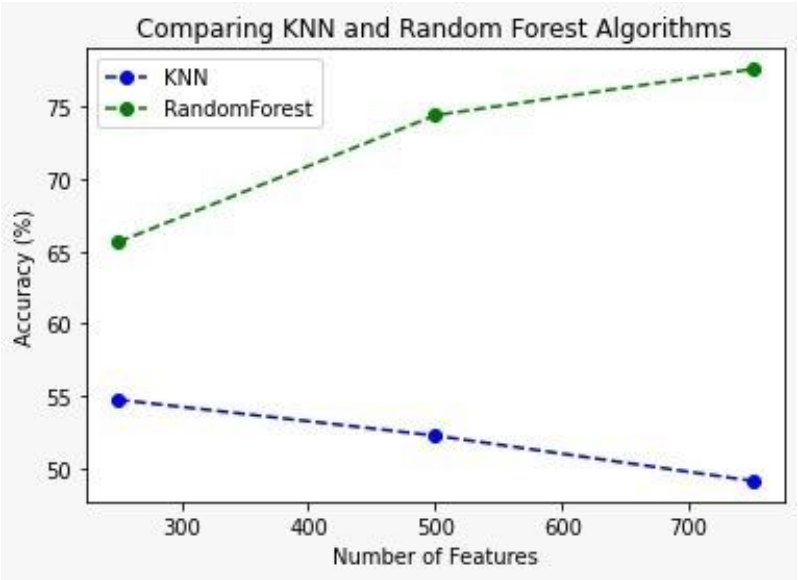

Fig. 2. Line graph Comparing kNN and RandomForest Algorithms

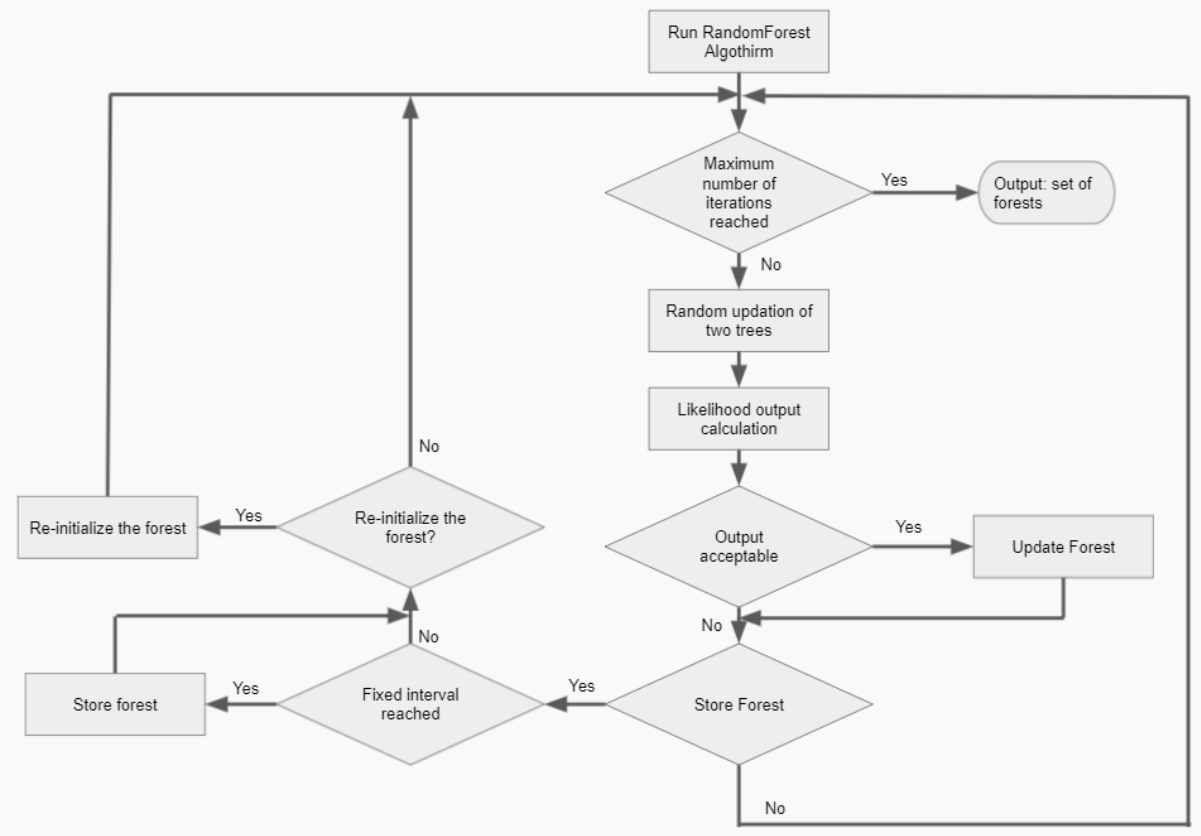

Fig. 3. Visual representation of the random forest algorithm

3.6 Determining public sentiment towards the agricultural ministry

We use the RF algorithm to perform the sentiment analysis in order to determine the public opinion on the Ministry of Agriculture. The dataset obtained from Twitter on the Ministry of Agriculture was now processed using the trained classifier to determine the public sentiment.

4. Experiment Results

We observed that the Random Forest Algorithm is better than the k Nearest Neighbours Algorithm. After using the Random Forest algorithm on the collected tweets, it is observed that the public opinion towards the Ministry of Agriculture of the Government of India is 48% negative, 35% positive and 17% neutral. Hence, we conclude that the public opinion towards the Indian Ministry of Agriculture is mostly negative. This could be owing to the backlash against the Farm Bills passed in September 2020.

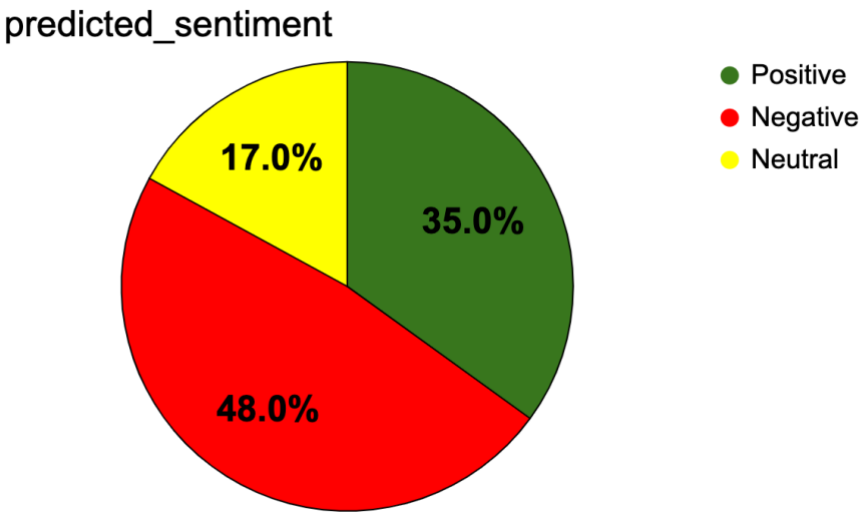

Fig. 4: Pie Chart depicting public sentiment towards the agricultural ministry

5. Discussion

The decrease in accuracy observed in the k Nearest Neighbours (kNN) algorithm is possibly an example of the Curse of Dimensionality defined by Bellman in his book Dynamic Programming. Under this, the increase in the dimensionality, that is, increase in the number of features for training the machine learning algorithms like kNN increases the sparsity of the dataset and thus reduces the accuracy of the algorithm. A few solutions to combat this may include dimensionality reduction and Principal Component Analysis [17].

6. Conclusion

This paper was used to compute the acceptance rate or the public sentiment towards the Ministry of Agriculture, Government of India. This analysis can help the Government in reforming the policies that cause a negative public opinion. We used two algorithms, namely Random Forest and k Nearest Neighbors, to determine which algorithm amongst the two is better able to find the sentiment of a text (tweets) by gradually increasing the number of features to be considered. We used the TFIDF vectorization method to convert textual data of tweets into numerical data. Furthermore, we observed that the Random Forest algorithm offered more accuracy when compared to the kNN algorithm. After performing Twitter Sentiment Analysis, we extracted the tweets about our use case which is regarding the Ministry of Agriculture of the Government of India using the Tweepy module. We found that the extracted tweets upon vectorization contained a total of 3,038 features. We accordingly trained the Random Forest Classifier and conducted the sentimental analysis for the tweets regarding the Ministry of Agriculture. We hence found that the public's sentiment towards the Indian Ministry of Agriculture is mostly negative, possibly because of the backlash due to the newly introduced agricultural reforms in the country.

7. Future Enhancements

This study can be extended to all the ministries under the Government and can result in creating awareness about the ministries that are underperforming according to the public's sentiment. Further, it can be extended to not only the central Ministries but also to various political parties to gather data regarding a particular party and how the people feel about it. As there is an increase in the data, the target dataset increases as well and will increase the accuracy of the algorithm. This study can also be implemented using other Machine Learning algorithms with more feature extraction to check for a better accuracy score.

REFERENCES

1. Vasudevan, Srividhya & Meenakshi, G.. (2018). Comparison of Sentiment Analysis of Government of India Schemes using Tweets. International Journal of Computer Sciences and Engineering. 6. 998-1001. 10.26438/ijcse/v6i6.9981001.

2. Sharma, Sujeet & Shetty, Nisha. (2018). Determining the Popularity of Political Parties Using Twitter Sentiment Analysis. 10.1007/978-981-10-7563-6_3.

3. Verma, Pankaj &Khanday, Akib & Rabani, Syed & Mir, Mahmood & Jamwal, Sanjay. (2019). Twitter Sentiment Analysis On Indian Government Project Using R. International Journal of Recent Technology and Engineering. 8. 8338-8341. 10.35940/ijrte.C6612.098319.

4. Naldi, Maurizio. (2019). A review of sentiment computation methods with R packages.

5. Vidya Zope, Anagha Karmarkar, Mansi Shivani, Vinit Pawar, Kanchan Tewani (2018). Analysis of Twitter Reactions to Government Policies. *International Journal of Advance Research, Ideas and Innovations in Technology*, 4(1) www.IJARIIT.com.

6. Chen, Hsuanwei& Franks, Patricia & Evans, Lois. (2016). Exploring Government Uses of Social Media through Twitter Sentiment Analysis. Journal of Digital Information Management. 14. 290-301. 10.6025/jdim/2016/14/5/290-301.

7. Rupinder Kaur, Rajvir Kaur, Manpreet Singh, Dr. Sandeep Ranjan. (2020). Twitter Sentiment Analysis of the Indian Union Budget 2020. *International Journal of Advanced Science and Technology*, *29*(4s), 2282 - 2288. Retrieved from http://sersc.org/journals/index.php/IJAST/article/view/10258

8. Pihlqvist, Fredrik and B. Mulongo. "USING RULE-BASED METHODS AND MACHINE LEARNING FOR SHORT ANSWER SCORING." (2018).

9. Naiknaware, B.R., Kawathekar, S., & Deshmukh, S. (2017). Sentiment Analysis of Indian Government Schemes using Twitter Datasets.

10. Pak, P. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA).

11. P. Mishra, R. Rajnish and P. Kumar, "Sentiment analysis of Twitter data: Case study on digital India," 2016 International Conference on Information Technology (InCITe) - The Next Generation IT Summit on the Theme - Internet of Things: Connect your Worlds, 2016, pp. 148-153, DOI: 10.1109/INCITE.2016.7857607.

12. Sravya, K., Sowmya, G., Yamini, P., Anusha, P., & Sandhya Krishna, P. (2021). Sentiment Analysis On Twitter K. The International journal of analytical and experimental modal analysis, 13, 925-930.

13. Yadav, N., Kudale, O., Rao, A., Gupta, S., &Shitole, A. (2021). Twitter Sentiment Analysis Using Supervised Machine Learning. In Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2020 (pp. 631-642). Springer Singapore.

14. Mohammad Rezwanul Huq, Ahmad Ali and Anika Rahman, "Sentiment Analysis on Twitter Data using KNN and SVM" International Journal of Advanced Computer Science and Applications(IJACSA), 8(6), 2017. http://dx.doi.org/10.14569/IJACSA.2017.080603

15. P. Karthika, R. Murugeswari and R. Manoranjithem, "Sentiment Analysis of Social Media Network Using Random Forest Algorithm," 2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS), 2019, pp. 1-5, doi: 10.1109/INCOS45849.2019.8951367.

16. Charan Gowda, Anirudh, Akshay Pai, and Chaithanya kumar A, "Twitter and Reddit Sentimental analysis Dataset." Kaggle, 2019, doi: 10.34740/KAGGLE/DS/429085.

17. Venkat, Naveen. (2018). The Curse of Dimensionality: Inside Out. 10.13140/RG.2.2.29631.36006.