iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** VII **Month of publication:** July 2020

DOI: https://doi.org/10.22214/ijraset.2020.30636

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com

Twitter Sentiment Analysis on Citizenship Amendment Act in India

Arohi Narang[1], Shambhavi Kumar[2], Monica Parihar[3]
[1, 2, 3]*Student, EXTC, MPSTME, NMIMS University*

Abstract: ***Sentiment analysis is an opinion mining process, in which computational analysis and categorization of opinion of a piece of text is done to obtain an unbiased understanding of the writer's opinion towards any specific topic. In this paper, Sentiment Analysis of the twitter user demographic towards Citizenship Amendment Act, which came into effect in India from January 10th, 2020, has been done. CAA was considered, as it had garnered mixed opinions from different sections of the Indian demographic, so there was no clear understanding of the overall sentiment of the public towards it. It had also led to protests and riots in various parts of India, which the Government struggled to handle as it was unexpected. This paper presents a faster approach of sentiment analysis of a large demographic, by using various classifiers for categorization of opinion of data into positive, negative or neutral. VADER was used for faster and accurate POS tagging of data. Support Vector machine obtained the highest accuracy at 77.32% out of all the remaining classifiers.***

Keywords: Sentimental analysis, CAA (The Citizenship Amendment Act), VADER(Valence Aware Dictionary and Sentiment Reasoner), classifiers, KNN, logistic regression, SVM, Naïve Bayes, random forest, decision tree

I. INTRODUCTION

Sentiment Analysis refers to the process that helps an organization extract information about how their clientele or any group of people is reacting to a particular product, service or a newly launched policy. In essence, Sentiment Analysis alludes to the utilization of natural language processing (NLP), text mining, computational linguistics, and bio measurements to methodically recognize, extricate, evaluate, and examine emotional states and subjective information as gathered correctly in the paper [4]. Natural Language Processing basically has the objective to understand and create a natural language by using necessary tools and techniques. On Twitter, users can share their opinions in the form of tweets within 140 characters.

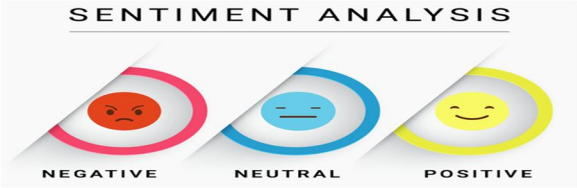


Figure 1: Classes used in Sentimental Analysis [21]

The above picture shows the three classes positive, negative and neutral. This leads to people abridging their statements by using slang, abbreviations, emoticons etc. Along with this, people also use sarcastic and polysemy language in their tweets. Hence it is common knowledge that Twitter language is unstructured. In order to retrieve meaningful information from tweets, sentiment analysis is used which gives result in terms of total number of tweets that positive, negative and neutral. The results are obtained using various classifiers and the most accurate classifier is taken into consideration. These results can be effectively utilized for various purposes including analysing general reaction of citizens towards a newly drafted policy or a recently launched act, a classic example being the Citizenship Amendment Act which will be further discussed in this paper. Citizenship Amendment Act was proposed as a bill on 9 December 2019 in 17th Lok Sabha and passed by the Rajya Sabha on 11 December 2019. The Citizenship Act aims to amend the definition of illegal immigration from neighbouring countries including Pakistan, Afghanistan and Bangladesh who have lived in India without documentation. They will be granted fast-tracked citizenship in 6 years although the minimum requirement of stay for a foreigner is 12 years. Although there is no mention of the word 'Muslim' many petitioners say the act discriminates against Muslims and violates the fundamental Right to Equality. Protests have broken out across India, a few of them violent following the launch of this act. Sentimental analysis can be very useful for future drafting of new government policies. Before amending any Acts or introducing any new Laws the government can first put the idea forward unofficially on various platforms so as to understand the sentiments of the public towards it and accordingly make the policies more relevant. Hence this paper, we have proposed to perform sentimental analysis on Citizenship Amendment Act using Twitter database.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com

II. LITERATURE SURVEY

Through the years, several papers have been published on Sentiment Analysis. Various methods and approaches have been observed in these papers for implementation of Sentiment Analysis. From the works available, this survey presents some of these approaches available in literature:

In Large-Scale Sentiment Analysis for News and Blogs by Namrata Godbole [5], Lydia text analysis system was used to determine the sentiments of the people. Lydia Text Analysis System is still at a very early stage of development. In the following model instead of Lydia a better lexicon dictionary can be used that can identify emoticons, slang words and can give a separate rating for small and capital letter for higher accuracy.

In the paper [7] Hassan Saif, semantics is added as additional feature into the training set for sentiment analysis. Using the interpolation approach the semantic features has been fed into Naıve Bayes (NB) model training. However the Naïve Bayes approach has many limitations such as data scarcity and zero frequency. Thus other classifiers should be used for better accuracy.

Analyzing Awareness of Government Scheme using Swachh Bharat Tweets by Pooja Dhede[1]. This model uses Twitter as its database and shows if the proposed government scheme will have good/bad or positive/negative impact on the society. For higher accuracy a large number of tweets need to be accessed.

Twitter Sentiment Analysis of Movie Reviews using Machine Learning Techniques by Akshay Amolik [3]. This model uses feature vector and classifiers such as Support vector machine and Naïve Bayes to perform sentiment analysis of tweets with of latest reviews of upcoming Bollywood or Hollywood movies. The tweets are then correctly classified as positive, negative and neutral.

In sentiment analysis of demonetization of 500 & 1000 rupee banknotes by Indian government by Prabhsimran Singh [2], the geolocation feature has been used to make a nation and state wide analysis of reasons of displeasure among people towards this government policy.

III.SYSTEM DESCRIPTION MODEL

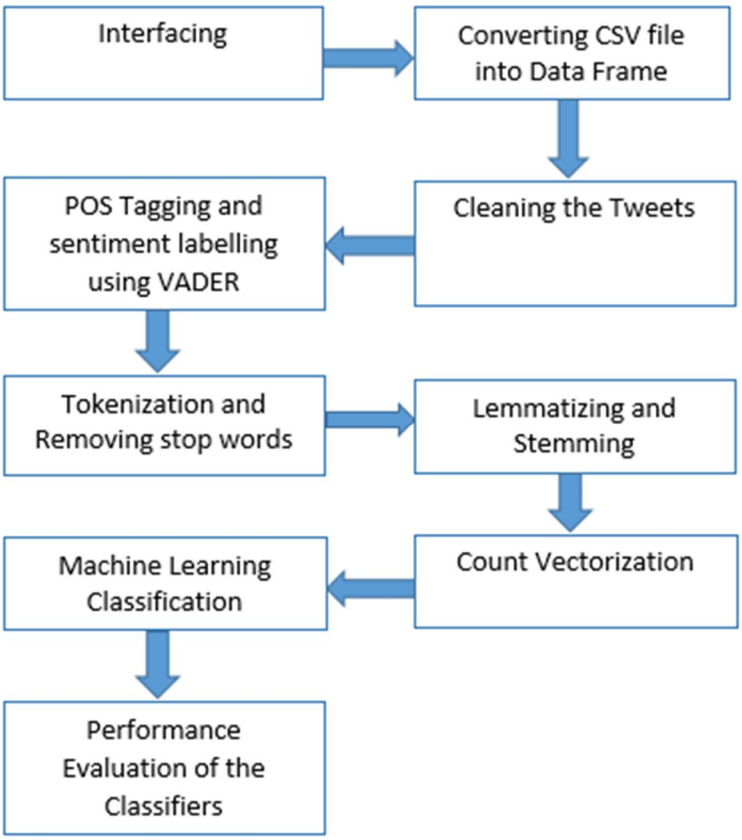


Figure 2: System Model flowchart

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com

A. Collecting Data

Data can be collected by using various API's such as Reddit API, Facebook API, Twitter API, and scraping data from web pages. We decided to go with Twitter API as twitter is considered the "Gold Mine of Data". Unlike other social media platforms, almost every user's tweets are completely public and extractable which provides a large database for analysis as mentioned in [6]. A. Shelar in the paper [19] shows how Tweepy is a python client for the official Twitter API. It is a wrapper over the raw Twitter API and provides a lot of heavy lifting for creating API URLs and http requests. Only the API keys and tokens need to be provided from the Twitter developer account and Tweepy takes care of talking with Twitter API. Thus to access the tweets, one has to create a twitter-developer account as mentioned in the papers [4,5]. The Twitter team will thoroughly review the application. After this step an app is created to get access to API key, API secret key, Access secret token which is used to login through the portal. After the authentication, tweets are downloaded from the Twitter database using the keyword "#CAA", after which, all the tweets are saved in a text file.

B. Converting useful data to CSV

The text file is converted to CSV where only the relevant objects are kept such as objects containing the entire text of the tweet, retweet count of a tweet, geographic location of a tweet, latitude and longitude coordinates of the place, follower count of the user who tweeted, creation time and date of tweet and id of the tweet in a string format. Rest of the objects are discarded.

C. Dropping Rows and Columns

There are two types of duplicates, one which has the same values for all columns and the other having repetitive text for tweets. Column containing the geographic location of the tweet is empty for most rows whereas column containing user id information and his follower count are not of much use so all these unnecessary rows and columns are deleted.

D. Cleaning the Tweets

This step removes special characters (such as hashtags, back slash, brackets, etc), numbers, punctuation, URLs, and replaces them with white spaces.

```
df["full_text"][0]
```

```
'  You shamelessly abandoned ur Sikh brothers to the devil to please ur Madam
```

Figure 3: Result showing a cleaned tweet after removal of special characters and URL's

E. POS-Tagging and Sentimental Analysis using VADER

In Part-of-Speech Tagging, each token or word present in a sentence, is assigned a label which indicates various grammatical categories like singular/plural, tenses, the part of speech, etc.

This has been implemented by using the Valence Aware Dictionary and Sentiment Reasoner (VADER) module, demonstrated step-by-step approach in [20, 6].

Sentimental Analysis is basically statistical analysis of a piece of text so as to work out whether it is negative, neutral or positive. Sentimental Analysis is mostly done by taking any of the two approaches: Valence-based or Polarity-based. In polarity-based approach, the text is simply classified as positive or negative. This means that 'disastrous' and 'bad' will be considered to have the same sentiment i.e. negative. On the other hand, in valence-based approach, the intensity of the sentiment is also taken into consideration, i.e. 'disastrous' is considered more negative than 'bad'.

VADER takes a valence-based approach for analysing a piece of text.

VADER analyses sentiments of texts, based on lexicons of sentiment-related words. It analyses a text and checks to see whether any of the words present in the sentence is present in its lexicon dictionary. It accordingly rates them positive, negative or neutral [8].

Not only does VADER rate the sentences based on the words present in it but it also focuses on the capitalization of the words and the sentence construct.

For example, say the sentence "Today's weather is good and I am feeling excellent" is considered. In the sentence, 'excellent' and 'good' are rated 1.95 and 1.7 respectively. But if the word good was capitalized then the rating given would be more. VADER also rates the sentence based on any exclamation mark or emoticons present in the sentence. Which makes it ideal for use on social

©IJRASET: All Rights are Reserved

media data. Not only that, but it also takes into consideration the use of modifying words, if any, like 'extremely', 'very', 'too' and so on, before a sentiment term. For example, "kinda bad" would result in a decrease in the negative intensity of a sentence, but "extremely bad" would result in an increase in the negative intensity of a sentence.
Another feature of VADER is that it can also handle changes in a sentence's sentiment intensity when it contains 'but'. According to the rule, the sentiment of the sentence before and after 'but', both is taken into consideration, but the sentiment for sentence coming after 'but' is weighted more than the one coming before it.

```
In [132]: df3.comp_score.value_counts()

Out[132]: Negative    1293
          Positive    924
          Neutral     626
          Name: comp_score, dtype: int64
```

Figure 4: Result showing number of sentences classified into classes Positive, negative and neutral after POS-Tagging

F. Tokenization

Tokenization is the process of breaking up a character sequence or a sequence of string into phrases, words and keywords called tokens. The tokens are obtained by splitting the sequence by white-spaces, hyphens or apostrophe, depending on the need, and at the same time removing characters like punctuation marks. The tokens then become input for the next step.

G. Stop-word Removal

Stop words are commonly used words such as "the", "an", 'in" etc. which are deemed irrelevant as they occur frequently in the sentences and do not add any significant meaning to the sentiment. They increase the size of the data unnecessarily.

H. Lemmatization and Stemming

The goal of both stemming and lemmatization is to reduce phonetic forms and derived words to its base word. For example:

am, are, is $\Rightarrow$ be

shoe, shoes, shoe's, shoes' $\Rightarrow$ shoe

The resultant of the text will then be:
The girl's shoes are different colors $\Rightarrow$ The girl shoe be differ color

Stemming usually refers to a raw process that removes the suffix of words, which often includes the removal of derivational affixes. *Lemmatization* is a more accurate approach which does vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the *lemma* .It is important that the above three cleaning processes (Stop words Removal, Stemming, Lemmatizing) are not performed before (sentiment labelling) because doing so will cause irregularities in sentiment detection.

I. Count Vectorization

It is a tool in Scikit-learn library of python. Count vectorization is the transformation of any given text into a vector, based on the frequency count of occurrence of any word in the text. It uses two features:-

1) min_df that defines minimum frequency of a word to be used as a feature
2) ngram_range which is a tuple.It defines the minimum and maximum length of sequence of tokens considered. As seen in Figure 5 the n-gram is (1,1) so this finds sequence of 1 token and the min_df is 4.

For example let's say a given document contains the following texts:
text1: Your dog is smart
text2: A cat is not a dog
text3: This is my dog

©IJRASET: All Rights are Reserved

Then the bag of words obtained for the given document where only the identified unique words are included after converting to lowercase will be:-
['the', 'dog', 'is', 'smart', 'a', 'cat', 'not', 'this', 'my']
This bag of words is then sorted. The CountVectorizer function maps each word to feature indices as shown below in Table 1. Also, for each word, its count in a particular text sample is included in each cell using CountVectorizer function

TABLE 1
Table 1: Accuracy results for various classifiers

Indices->	0	1	2	3	4	5	6	7	8
Vocabulary->	a	cat	dog	is	my	not	smart	this	your
Text1	0	0	1	1	0	0	1	0	1
Text2	2	1	1	1	0	1	0	0	0
Text3	0	0	1	1	1	0	0	1	0

The CountVectorizer function produces the sparse matrix as shown below:

```
array([ [0 0 1 1 0 0 1 0 1],
        [2 1 1 1 0 1 0 0 0],
        [0 0 1 1 1 0 0 1 0] ], dtype=int64)
```

J. Machine Learning Classification

Classification is a process of building a model. Classification belongs to the category of supervised learning. A model can be thought of as a mathematical equation used to predict a value by giving one or more values to it. It relates one or more independent variables to dependent variables. The more relevant data and the more number of dependent variables we have the more accurate model we get.

In our model we have split our dataset into training dataset and test dataset by importing sklearn.model_selection. Sklearn is a python library that offers various features for data processing that can be used for classification, clustering, and model selection. Model selection splits the input data that could be in the form of lists, arrays or dataframes into random train and test datasets. The training set contains a known output and the model learns from this data in order to be generalized to other data later on. We have the test dataset (or subset) in order to test our model's prediction on this subset. We have then used the train_test split function in order to make the split. The test size is 0.2 that means that 20% of the total data is test data and remaining 80% is training data. The output of the function is stored in variables that are X_train, y_train, X_test, y_test.In our dataset X_train and X_test are the actual tweets and y_train and y_test is the sentiment to which the tweet belongs. Since a classifier cannot work on text data directly we need to convert it into a vector using CountVectorizer function. Thus X_train and X_test are converted into their vector forms and.X_train (vector) along with y_train are fed into the classifiers to train them. After training the model X_test (vector) is fed into the model and the model makes a prediction for every test data sent as input.

In our dataset we have three classes positive, negative and neutral. The following classifier model is trained and is fed with known positive negative and neutral tweets, as that is used as the training data. After the classifier is trained accurately, it can be used to detect an unknown tweet and can give its correct class automatically.

```
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
SEED=1
X = df3.full_text
y = df3.comp_score
#Using CountVectorizer to convert text into tokens/features
vect = CountVectorizer(stop_words='english', ngram_range = (1,1), max_df = .80, min_df = 4)
X_train, X_test, y_train, y_test = train_test_split(X,y,random_state=SEED, test_size= 0.2)
#Using training data to transform text into counts of features for each message
vect.fit(X_train)
X_dtm=vect.transform(X)
X_train_dtm = vect.transform(X_train)
X_test_dtm = vect.transform(X_test)
```

Figure 5: Splitting dataset into training and test data

©IJRASET: All Rights are Reserved

We use the Sklearn library to import the various classifier models. The various classifier models that we have used are as follows:-

1) *K Nearest Neighbour-* KNN is a Lazy Learner classifier, as it only makes use of training data for prediction of class. The training data is already labelled, through which it learns to label new points through similarity measure by using distance functions. It identifies the K nearest neighbours based on the distance function. Various distance functions that can be used are Euclidean distance, Manhattan distance, and Minkowski distance. These three are used for continuous variables. Hamming distance is used for categorical variables.
2) *Naïve Bayes:* It is known as generative learning model which is derived from Bayes Theorem. In this model features of different classes are assumed to be independent of each other irrespective of their actual dependency on each other. All these features independently contribute to the probability. Naïve Bayes is useful for analysing large data sets, and is easy to implement. Accuracy results that can be obtained is satisfactory given its simple model and the amount of data that can be handled using Naïve Bayes as clearly demonstrated in [13,14].
3) *Support Vector Machine:* Mullen. [9] and Muscolino. [10] Showed that in SVM, classification of data item is done based on its position in n-dimensional space with respect to the hyperplane. As shown in Figure 6, individual observations are plotted. Support vectors are simply coordinates of these observations. Support vectors from each class that are closest to each other are then selected. For each selected vectors the margin (distance) from all the possible hyperplane is computed. The hyperplanes with the maximum margin is the optimal hyperplane which differentiates the various classes the most. Based on position of these hyperplanes, classification of new data items is done.

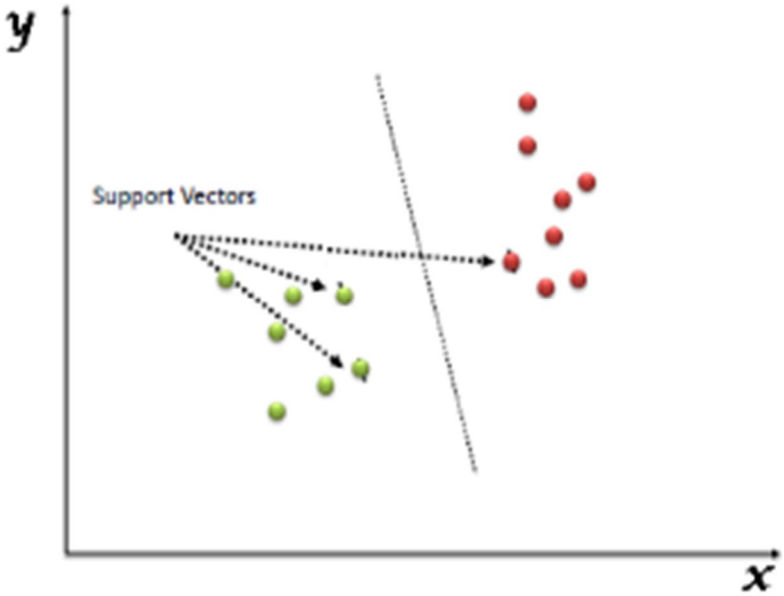


Figure 6: Example showing distribution of Support Vectors [22]

4) *Random Forest:* It is an ensemble algorithm, i.e. it combines more than one algorithm (Eg. SVM, KNN, etc) for classifying objects. It breaks a dataset into smaller subsets and creates a decision tree by combining them. It decides final class of input data by aggregating the votes from different decision trees [17]. This reduces noise thus giving more accurate results.
5) *Logistic Regression:* It is an analytical method of classification or a predictive learning model. It analyses data sets which contain one or more dependent variable that determine an outcome. In this dichotomous variables are used, i.e. variables that have only two possible outcomes, to predict the outcome as mentioned in [8]. Logistic regression finds the best fitting model that describes the relation between dichotomous characteristic of interest and a set of independent variable. As it quantitatively reasons the factors that lead to classification, it is better than KNN.
6) *Decision Tree:* Decision tree is a classification or regression model. It works on categorical and numeric data. In this, the dataset is broken into smaller subsets and simultaneously a tree structure formed. The tree consists of various nodes that is decision nodes, leaf nodes and root node. Each leaf node represents a classification or decision and each decision node is divided into two or more branches. The root node is the topmost decision node and corresponds to the best classifier.

©IJRASET: All Rights are Reserved

K. Performance Evaluation of Classifiers Model

After training the following classification algorithms it is important to know which algorithm works best for our dataset that is which one has the most accurate predictions.

The classification report is used to measure their quality of prediction. The report shows us the precision, recall and F1 score for every class.

A confusion matrix gives us information about the true positives, false positives, true negatives and false negatives [23].

1) *TN (True Negative):* when a case was negative and predicted negative
2) *TP (True Positive):* when a case was positive and predicted positive
3) *FN (False Negative):* when a case was positive but predicted negative
4) *FP (False Positive):* when a case was negative but predicted positive

a) *Precision:* It is defined as the accuracy of the positive predictions or the ability of a classifier not to label an instance positive that is actually negative [23]. For each class:

Precision =correctly predicted class/total predictions for that class by the model

Or

TP/TP+FP

b) *Recall*: It is defined as the fraction of positives that were correctly identified or the ability of a classifier to find all positive instances [23]. For each class

Recall=correctly classified class/actual number of instances of that class in dataset

OR

TP/TP+FN

c) *F1 score*: It is defined as the weighted harmonic mean of precision and recall. It ranges from 0.0 to 1.0. [23]

For every class F1 score is

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

d) *Support*: It is defined as the number of actual occurrences of the class in the specified dataset.[23]

e) *Accuracy:* It is defined as the ratio of the total correct predictions made to the total number of predictions made. In most real-life classification problems, imbalanced class distribution exists and thus **F1**-score is a **better** metric to evaluate our model. [23]

We also get to know about the weighted average precision, weighted average recall and weighted average F1 score.

Let's say we have three classes i.e class A, B, C

Weighted average precision=

[(TP of class A × precision of class A) + (TP of class B × precision of class B) + (TP of class C × precision of class C)] ÷ (Total number of occurrences of all classes in the dataset)

Weighted average recall=

[(TP of class A × recall of class A) + (TP of class B × recall of class B) + (TP of class C × recall of class C)] ÷ (Total number of occurrences of all classes in the dataset)

Weighted average F1 score=

[(TP of class A × F1 score of class A) + (TP of class B×F1 score of class B) + (TP of class C×F1 score of class C)] ÷ (Total number of occurrences of all classes in the dataset)

IV. RESULTS AND DISCUSSION

A. Data Set

Twitter data set was used which was obtained using Twitter API. Around 10,000 tweets were accessed from Twitter for training the classifiers.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com

B. Performance Evaluation

TABLE 2
Table 2: Accuracy results for various classifiers

Classifier	Accuracy	Weighted Average Recall	Weighted Average Precision	Weighted Average F1 score
Naïve Bayes	64.14	0.64	0.64	0.63
Support Vector Machine	77.32	0.77	0.78	0.77
Logistic Regression	76.80	0.77	0.77	0.77
K Nearest Neighbours	49.03	0.49	0.66	0.49
Decision Tree Classifier	72.75	0.73	0.74	0.73
Random Forest Classifier	76.09	0.76	0.77	0.76

TABLE 3
Table 3: Accuracy results for each class

Classifier	Classes	Recall	Precision	F1 score
Naïve Bayes	Negative	0.73	0.71	0.72
	Neutral	0.34	0.61	0.43
	Positive	0.70	0.54	0.61
Support Vector Machine	Negative	0.80	0.85	0.82
	Neutral	0.77	0.70	0.73
	Positive	0.72	0.71	0.71
Logistic Regression	Negative	0.80	0.84	0.82
	Neutral	0.76	0.71	0.73
	Positive	0.71	0.69	0.70
K Nearest Neighbours	Negative	0.39	0.81	0.52
	Neutral	0.96	0.33	0.49
	Positive	0.34	0.62	0.43
Decision Tree Classifier	Negative	0.75	0.82	0.78
	Neutral	0.76	0.68	0.72
	Positive	0.66	0.62	0.64
Random Forest Classifier	Negative	0.78	0.84	0.81
	Neutral	0.79	0.66	0.72
	Positive	0.70	0.71	0.70

We can observe the accuracy of each Classifier in Table 2 and Table 3. The accuracy value for each classifier shows the percentage of test data that the classifier was able to classify correctly.

©IJRASET: All Rights are Reserved

On running the code, we compared the output of every classifier for the first five test inputs with the outputs we got on using VADER.

As we can see in the table, SVM gives us the highest accuracy and KNN the least. Below we will compare the predicted outputs for both the classifiers on the same test data.

The first five test data with their sentiments as predicted by VADER are given in the Figure 8 below:

	original_text
1929	Dr. Kafeel khan is a man of dedication, being a doctor he stepped as high for doing social work regardless of religion, but for humanity.He is jailed bcs he stood against the black bill #CAA #NRC and fascist government\n#FreeDrKafeel https://t.co/TgnqrzqLeG
1131	@JoeBiden Mr.Joe, you're against #CAA without knowing actual reason for it. Time & again, since #Decades #minorities #Hindus have been #Assaulted #Killed #Converted #HolyPlaces demolished by #PEACEFULS including #Christians \n#CAA protect their #survival #race \n#CAA 4 Minorities https://t.co/fovnCrN1LA
2447	@IfraJan_ Hence #CAA https://t.co/9eqlAjR253
1872	Genesis of #DelhiPogrom emanates from provocative speechesâ€ intolerable to civilised societyâ€ given by politicians to target anti-#CAA protestors, not the other way around: SC lawyer M R Shamshad, head of Minorities Commission inquiry https://t.co/yNcGj87F0B
2034	Hypocrisy: the practice of claiming to have moral standards or beliefs to which one's own behavior does not conform; pretense. #JJAbrams @jjabrams #Disney @Disney @DisneyStudios #CAA @caaspeakers @caafoundation When I speak out, JJ rejects me for it, then says #BelieveSurvivors

Figure 7: Text data before pre-processing

full_text	compound	comp_score	SVM_Output	KNN_Output
Dr Kafeel khan is a man of dedication being a doctor he stepped as high for doing social work regardless of religion but for humanity He is jailed bcs he stood against the black bill CAA NRC and fascist government FreeDrKafeel	-0.8807	Negative	Negative	Negative
Mr Joe you re against CAA without knowing actual reason for it Time amp again since Decades minorities Hindus have been Assaulted Killed Converted HolyPlaces demolished by PEACEFULS including Christians CAA protect their survival race CAA Minorities	-0.7430	Negative	Negative	Negative
Hence CAA	0.0000	Neutral	Neutral	Neutral
Genesis of DelhiPogrom emanates from provocative speeches intolerable to civilised society given by politicians to target anti CAA protestors not the other way around SC lawyer M R Shamshad head of Minorities Commission inquiry	-0.3182	Negative	Negative	Negative
Hypocrisy the practice of claiming to have moral standards or beliefs to which one s own behavior does not conform pretense JJAbrams Disney CAA When I speak out JJ rejects me for it then says BelieveSurvivors	-0.1197	Negative	Neutral	Neutral

Figure 8: Sentiment as predicted by VADER (comp_score), SVM and KNN for first 5 text input

©IJRASET: All Rights are Reserved

```
Confusion Matrix:
[[235  23  35]
 [ 15  91  12]
 [ 28  16 114]]
```

```
Confusion Matrix:
[[113 150  30]
 [  2 113   3]
 [ 25  80  53]]
```

Figure 9: Confusion matrix of SVM (left) and KNN (right)

As seen in Table 2, the accuracy of SVM is 77.32% and F1 score is 0.77. Its weighted average recall is same while precision is 0.78. That means the number of false positives and the number of false negatives predicted by the classifier model is almost same. On the other hand KNN has an accuracy of 49.03% and F1 score of 0.49. It has a high precision and low recall which means that the classifier's accuracy of positive predictions or the ability of a classifier not to label an instance positive that is actually negative is good but it's ability to find all positive instances is poor. Thus KNN will have more false negatives than false positives. Thus SVM is the most preferred model.

V. CONCLUSION AND FUTURE SCOPE

It has been concluded that the Support Vector Machine is more accurate than the other classifiers used. As it is an Eager Learner, Support Vector Machine's prediction time is also better than other classifiers. To get a better accuracy of the following classifiers one can access more number of tweets. Currently the model's accuracy has been checked by accessing 10,000 tweets but one can access more number of tweets. The above problem can be approached using deep learning techniques.

Since VADER already contains various slang words and emoticons in its lexicon, it is easier to use it for analysis of any social media database.

Automated Sentimental Analysis is one of the fastest approach an organization or a government can take for better analysis of the overall sentiments of any large demographic towards its policies, acts or amendments that it wishes to implement or pass in the near future. This will avoid riots, protests and loss of revenue as now there will be a better understanding of the demographics sentiments, which was not the case during the implementation of CAA in India.

VI. LIMITATION

Twitter sentiment analysis has a few disadvantages along with its many advantages. Primary limitation being words can have different contextual meanings in tweets, for example, the word 'devastating' usually has a negative connotation attached to it but in the context of music or art, it can also be used to imply an emotional engagement which is seen as positive. The other limitation is language barrier, when people use the English alphabet to express their tweets in native language, which has no dictionary yet to assign positive or negative values to the words. Sarcastic statements such as irony can also be misinterpreted leading to slightly inaccurate results. Sometimes tweets are accompanied by URLs and links to images and videos which are often an extension of the author's explicit opinion. In such a case, analysing the tweet solely on the basis of the 280 character limit permitted by Twitter can lead to unfair results. We can train the model by dividing the dataset into training, testing and validation dataset. We can train the model using the cross validation technique as that splits the dataset into several folds of training and testing. The model is trained on training dataset and tested on the testing dataset.

REFERENCES

[1] Pooja Dhede, Gaurav Chaudhari, Gaurav Gaikwad, Samruddhi Hagone, "Analyzing Awareness of Government Scheme using Swachh Bharat Tweets", VJER-Vishwakarma Journal of Engineering Research, June 2019.

[2] P. Singh, R. Singh Sawhney, K. Singh Kahlon, "Sentiment analysis of demonetization of 500 & 1000 rupee banknotes by Indian government", 2018.

[3] Amolik, Akshay & Jivane, Niketan & Bhandari, Mahavir & Venkatesan, "Twitter Sentiment Analysis of Movie Reviews using Machine Learning Techniques" (2016).

[4] A. Alsaeedi, M. K. Khan "A Study on Sentiment Analysis Techniques of Twitter Data" in (IJACSA) International Journal of Advanced Computer Science and Applications(2019).

[5] N. Godbole, M. Srinivasaiah and S. Skiena, "Large-Scale Sentiment Analysis for News and Blogs", 2007.

[6] V.K. Chauhan, A. Bansal and Dr. Amita Goel, "Twitter Sentiment Analysis Using Vader", International Journal of Advance Research, Ideas and Innovations in Technology, (2018).

[7] Saif, Hassan & Alani, Harith, "Semantic Sentiment Analysis of Twitter", 2012.

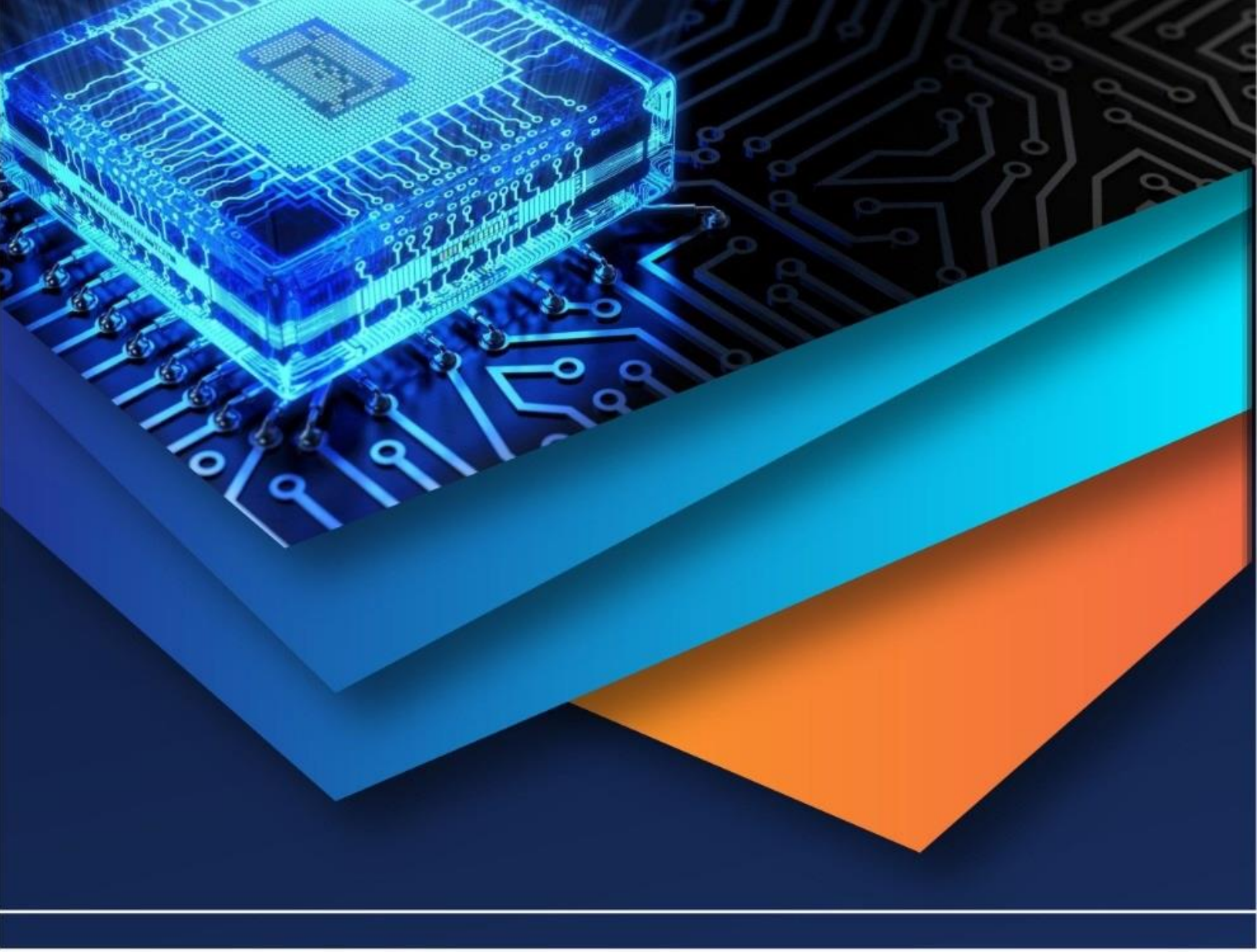
[8] A. Tyagi, N. Sharma, "Sentiment Analysis using Logistic Regression and Effective Word Score Heuristic" International Journal of Engineering & Technology, 7 (2.24) (2018) 20-23.

[9] Mullen, Tony & Collier, Nigel. (2004). Sentiment Analysis using Support Vector Machines with Diverse Information Sources.. 412-418.

©IJRASET: All Rights are Reserved

[10] Muscolino, Alessandro & Pagano, Salvatore. (2018). SENTIMENT ANALYSIS, A SUPPORT VECTOR MACHINE MODEL BASED ON SOCIAL NETWORK DATA. 10.15623/ijret.2018.0707020.
[11] D. Sharma, M. Sabharwal, "Sentiment Analysis for Social Media using SVM Classifier of Machine Learning", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue-9S4, Jul. 2019.
[12] P.B Matharasi, Dr. A.Senthilrajan, "Sentiment Analysis of Twitter Data using Naïve Bayes with Unigram Approach", International Journal of Scientific and Research Publications, Volume 7, Issue 5, May 2017.
[13] Dey, Lopamudra & Chakraborty, Sanjay & Biswas, Anuraag & Bose, Beepa & Tiwari, Sweta. (2016). Sentiment Analysis of Review Datasets Using Naïve Bayes' and K-NN Classifier. International Journal of Information Engineering and Electronic Business.
[14] K. Suppala, N. Rao, "Sentiment Analysis Using Naïve Bayes Classifier", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-8 June, 2019.
[15] H. Parmar, S. Bhanderi, & G. Shah, (2014), "Sentiment Mining of Movie Reviews using Random Forest with Tuned Hyperparameters".
[16] R. Prabowo, M. Thelwall "Sentiment analysis: A combined approach," Journal of Informetrics (2015).
[17] Suresh, Annamalai & Bharathi, C. (2016). "Sentiment Classification using Decision Tree Based Feature Selection". International Journal of Control Theory and Applications.
[18] Singh, J., Singh, G. & Singh, R. Optimization of sentiment analysis using machine learning classifiers. *Hum. Cent. Comput. Inf. Sci.* **7,** 32 (2017).
[19] A. Shelar, & Huang, Ching-Yu. (2018). "Sentiment Analysis of Twitter Data".
[20] S. Elbagir and J. Yang, "Twitter Sentiment Analysis Using Natural Language Toolkit and VADER Sentiment", Proceedings of the International MultiConference of Engineers and Computer Scientists 2019 IMECS 2019, March.
[21] 5 Things You Need to know about Sentiment Analysis and Classification from: https://www.kdnuggets.com/2018/03/5-things-sentiment-analysis-classification.html
[22] Understanding Support Vector Machine (SVM) algorithm from examples (along with code) from: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/
[23] Understanding the Classification report through sklearn from**:** https://muthu.co/understanding-the-classification-report-in-sklearn/

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)