GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

A NOVEL RANDOM FOREST IMPLEMENTATION OF SENTIMENT ANALYSIS

A Project Report of Capstone Project-2

Submitted by

ARNAV MUNSHI

(1613107014/16SCSE107055)

in the partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING WITH SPECIALISATION OF BUSINESS ANALYTICS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Under the supervision of

Sanchit Sapra , B.tech.,M.tech.,M.Sc.,

Assistant Professor

April/May-2020

GALGOTIAS
UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report "**SENTIMENT ANALYSIS OF TWITTER DATA USING RANDOM FOREST AND NLTK**" is the bonafide work of "**ARNAV MUNSHI**" who carried out the project work under my supervision.

SIGNATURE OF HEAD	**SIGNATURE OF SUPERVISOR**
Dr. MUNISH SABHARWAAL	SANCHIT SAPRA
PhD (Management), PhD (CS)	B.tech(CS),M.tech(CS),M.Sc(OR)
Professor & Dean,	**Assistant Professor**

School of Computing Science & Engineering

School of Computer Science & Engineering

ABSTRACT

In today's world, we humans have been communicating with each other through calls, social media applications like WhatsApp, Facebook, Twitter, etc. From the social media apps, we get social media data from those applications and check what sentences are positive and negative sentiment using sentiment analysis and using deep learning methods like deep neural networks for classifying them under positive or negative sentiment polarity from twitter accounts.

The data that we get from these social sites are being used for many social problems and used in government to analyze the opinion about social media users. This technique is called sentiment analysis.

The main purpose of this sentiment analysis for this project will be to comparatively determine the writings made by user and check if they are going towards positive or negative.

Only one technique will be used here – Machine Learning Algorithms –Random Forest. Random forest is being considered as one of the popular algorithms used by many people from big companies to data scientist. This algorithm is the proposed algorithm and this algorithm is being used by many existing authors working on the Sentiment Analysis on various fields .

Not only the Random Forest will be alone used , but we will be using another of the advanced Deep Learning Library used mainly for Text Analysis which is NLTK(mainly a library for Natural Library) will also be used here to work on the feature sets which will be explained in detail as we scroll down in this report.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

Motivation

The motivation behind of using Twitter Sentiment Analysis for my project was to observe a large number of tweets made in public and analyze them with reference to their native languages . As for Indian native language , Hindi , I chose to do the Sentiment Analysis based on the Hindi Tweets made by various people which I will be showing in later topics . Other than this , I found this project motivating because I found that the dataset is larger in twitter as compared to other sites . Secondly , not only Sentiment Analysis is used for text analysis or tweets analysis but also being used in various fields like : Various companies are selling their product online and promoting their products and services in social media platforms also . Now the customers might view those ads and click it to see about the product and if it looks interesting , then they go for checkout and order the product as per their interest . Now the company uses this Sentiment Analysis for analyzing the customer opinions about their product by seeing their comments , or filling a Q&A , reviews and so on and that's how they are able to know about the customer's likes and dislikes and that's how the companies grow .

Now , if I move on with my topic which is doing Sentiment Analysis on Twitter Data then it means for looking of the positive and negative tweets . This kind of work goes well in English but I am using Hindi Tweets dataset for this finding of this Positive and Negative tweets and using of Random Forest to find the score of my implementation which helps me to understand that how well this concept of Sentiment Analysis works on Twitter Data which may help in analyzing social threats by seeing over customer tweets and so on which may help the government to work on the issues . Although this kind of work has been done by many of the authors before me and I am also performing this project with only Random Forest and using only NLTK and see how my result stands out among the authors.

Domain Introduction

Now before this , we talked about the motivation and slight introduction towards Sentiment Analysis on Twitter Data. In this subtopic , I will talk somewhat a brief introduction about Twitter Sentiment Analysis.

Sentiment Analysis is a concept which falls under 'Pattern Classification' and Data Mining . But if we go towards in performing towards Sentiment analysis on Text Analysis like in Twitter data for analyzing tweets , then we also approach towards Text Mining and also in more advanced concept called Natural Language Processing.

Natural Language Processing (NLP) is defined as a field/concept in AI giving machine to read , understand and derive meaning from human native languages and in more simpler means handles human's natural language like speech/text.

The main application of NLP correspond to predictive of diseases , acting as a cognitive assistant , stopping spam mails by identifying the mails , helping in identifying fake news and also used in finance to generate massive reports.

For mainly to keep in concise , NLP also helps in sentiment analysis by identifying and extracting information from social media and provide lot of information about customer choice and decision drivers.

Now we will talk about NLTK which we are using as a NLP library in this project.

NLTK is one of the powerful NLP libraries and mainly used in Python to work with human language data. The NLTK provides a suite of Text processing libraries for classification , tokenization , stemming , tagging , parsing and semantic reasoning , wrappers for industrial-strength NLP libraries with the NLTK , we can do simple things , tokenization and tagging of text , naming entities and displaying parse trees.

There are about more than 100 micro blogging sites in today's world being used my millions of users across the globes through short messages and post. Like other micro-blogging sites , Twitter is also one of them where users daily post and share messages which is about millions of posts daily on an average and also comment about positive or negative views on a post made by someone. This kind of nature of humans making positive and negative comments are being supervised by many of the manufacturing companies where they see this as a way of earning profit regarding to the products and services and how the customer interacts with their products and services and this is how positive and negative comments and sentences or words got differentiated and studied leading to a new concept called SA(Sentiment Analysis).

Sentiment Analysis , being a study of the positive and negative comments , sentences or even words and these techniques are now being used by many of the Companies to know about their product well like Amazon , Google etc and so the list goes on.For this particular paper , I have used sentiment analysis on the Hindi tweets dataset and checked that how are the sentences being classified into positive , negative or neutral sentiment .

This can be achieved by using method – Machine Learning. ML - The best Machine Learning Algorithms for text classification will be Random Forest.

Random forest or Random Decision forest are the algorithms used for Classification , regression consist of a large number of individual decision trees and each individual tree comes out with a class prediction.

Random Forest Classifier- Random Forest defined as a large number of decision trees acting as a multiple machine learning algorithm to get better predictive performance as compared to using one decision tree algorithm. There will be more than one Decision Tree acting as a class prediction on an individual level and the class prediction's output generated individually make up the model prediction output for the Random Forest. The best part of Random Forest Classifier is that it can be used for Classification and Regression.

EXISTING SYSTEMS

Literature Survey

Sentiment Analysis on Twitter Data has been done and executed successfully by many authors which help us in getting more information about what the previous authors did is explained below:

1. Appoorv Aggrawaal presented Sentiment analysis of Twitter Data , used two combination of models like:

   a. Unigram model

   b. Tree Kernel Model

   c. 100 Sentiment features model etc.

   The author used Support Vector Machine and report averaged 5 fold cross validation test results and also used a binary classification task with positive and negative polarity and chance of baseline occurred to be 50 percent . He also investigated 2 kinds of models: tree kernel and feature based models and founded that these two model outperform the unigram model.

2. Duyu Tang , Furu Wei(2014) developed a Deep Learning System , called "Coooll" which is a deep learning model that builds sentiment classifier from Tweets and manually annotated sentiment polarity.

   The author used two kinds of features which are : SSWE feature and STATE features. As proposed by the author , the SSWE feature learned from 10 Million Tweets consisting of positive and negative emoticons and been verified in positive and negative classification of Tweets.

3. Varsha Sahayak , Vijaya Shete (Jan 2015) has proposed sentiment analysis on Twitter Data where the author uses Maximum Entropy method , Naïve Bayes Classifier and Support Vector Machine for Sentiment Analysis and did the following steps:

   - Retrieval of Tweets

- Pre-processing
- Parallel Processing
- Sentiment Scoring Module
- Output Sentiment

The author concluded that three Machine Learning Algorithms used by him outperforms the model namely , unigram , feature based model , Kernel model using WEKA.

He also concluded the difficulty increases with nuance and complexity of opinion expressed.

4. Dimitris Effrosynidis (21st Conference on Theory and Practice of Digital Libraries) which used pre-processing techniques like :

    a. Removing Numbers.

    b. Replacing repetition of punctuation.

    c. Handling Capitalized words.

    d. Lower-Casing .

    e. Replace slang and abbreviation.

    f. replace elongated word

    g. Replace contraction

    h. Replace negation

    i. Removing Stop words

    And using of advanced method like:

    a. Stemming

    b. Lemmatizing

    c. Replace URLs and user mentions

    d. Spelling correction

    e. Remove punctuations

5. Kalaivani A , Thenmozhi D (International Journal of Recent Technology , April-2019) only described that sentiment analysis has 2 categories of techniques which are Machine Learning Approach and Deep Learning Approach.

The Machine Learning approach comprises of 4 stages:

- Data Collection
- Pre-processing
- Training data
- Classification as well as plotting result

The feature of Machine Learning approach namely unigram , bigram , trigram and used algorithm like Support Vector Machine and Naïve Bayes.

The Deep Learning approach on sentiment analysis is very much efficient because they deliver impressive performance in NLP application and they don't need to be handpicked by doing the work themselves.

Every single unit in Neural Network is simple and by stacking layer of NN units of one competent to learn highly sophisticated decision boundaries and Algorithms like RNN are also competent in Sentiment Analysis.

6. Adyan Marendra Ramdhani(2017) proposed Twitter Sentiment Analysis using Deep learning method where the author used Deep Neural Network with including the specification of ReLU activation function , 3 Hidden Layers , Feed-forward Neural Network and using Mean Square Unit and Stochastic Gradient and preprocessing techniques like I mentioned earlier.

   The author concluded that Deep Neural Network achieved result of 75.03% and 77.45% and for MLP accuracy got 67.45% for train data and 52.05% on test data.

PROPOSED SYSTEM

In the previous topic , we looked at various methods used by authors ranging from Random Forest to Neural Network and now we have seen the existing models used by existing authors , Support Vector Machine was the Machine Learning algorithm mostly used and for this we will be using Random Forest so that we can compare our result among other existing results.

Below is the table of comparison shown for the results for existing models which we specified earlier in previous topic.

Author and Year	Method Applied	Purpose	Dataset Used	Result
Apoorv Aggrawal[2]	Support Vector Machine	Sentiment Analysis	Twitter Data	60.50%
Duyu Tang, Furu Wei(2014)[3]	Developed “Cooll” Deep Learning System	Sentiment Analysis	Twitter Data	70.14%
Varsha Sahayak , Vijaya Shete (Jan 2015)[4]	Maximum Entropy Method , Support Vector Machine , Naïve Bayes Classifier	Sentiment Analysis	Twitter Data	Sentiment classified into positive , negative and neutral
Kalaivani A , Thenmozhi D-[1]	SVM , KNN and DBN	Sentiment Analysis	Twitter Data	SVM(0.72 accuracy with n=0 and 0.62 with n=10), KNN(0.64 accuracy with n=0 and 0.63 with n=10), DBN(0.6 with n=0, 0.73 with n=10)
Andreyan Ramdhani(2017)[6]	Deep Neural Network , MLP	Sentiment Analysis	Twitter data	Deep Neural Network- 75.03% and MLP- 52.03%

Table 1: Summary of results by previous authors

Now below are the steps for the proposed system that we are using in this project and will also some explanation about each step.

Step 1: Downloading Hindi tweets and translating to English Tweets dataset:
In this step we are using Github link to download the dataset in Hindi tweet dataset. Later on, we convert add another column of Translated English tweet from Hindi Tweets and then import using Pandas library.

Step 2: Data visualization and Analysis:
Data visualization will be done on the dataset using matplotlib library in Python and analyze the data by cleaning by removing noisiness in them.

Step 3: Removing Usernames , stop-words , punctuations ,symbols :
In this step , after we have visualized the data in graph using matplotlib and seaborn library in python , we will be now doing the preprocessing of the dataset by removing

user name , stop-words , Punctuations ,symbols etc. to use the dataset in proper format and also using methods like stemming , lemmatization and tokenization which are techniques used in NLTK.

Step 4: Using Features extraction methods – BOW ,TF-IDF:
These methods – Bag of Words , TF-IDF will be used for creating features from text and will help in splitting the dataset into train and test dataset .

Step 5: Model Building –Random Forest Classifier:
In this step , after we have preprocessed the data in the proper format , we will be building the model using Random Forest Classifier. Random Forest Classifier is being defined as a large number of decision trees acting as a multiple machine learning algorithm to get better predictive performance as compared to using one decision tree algorithm acting as a class prediction on a individual level and class prediction's O/P individually make up votes and most votes with output make the model prediction O/P for Random Forest. The best part of Random Forest is that it can be used for Regression and Classification problem.

Below is the Architecture for the Random Forest Classifier :

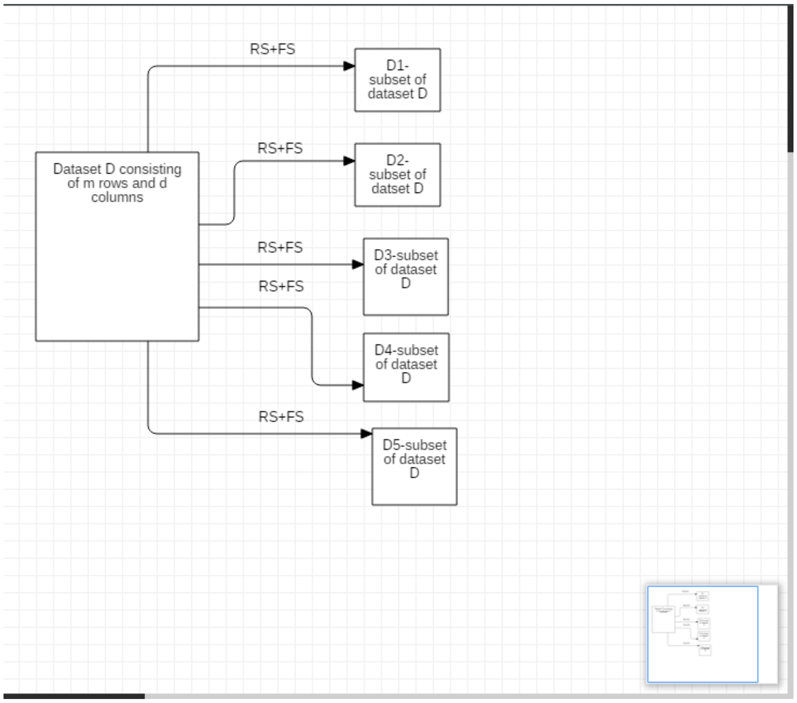


Figure 1: Architecture of Random Forest.

Step 6: Result with accuracy score and f1 score:
After the model is being fit using Random Forest Classifier and testing for prediction , we will use the accuracy score and f1 score.

With this we have explained the Proposed System and explained the some concepts.

IMPLEMENTATION

Now in previous section we mentioned the proposed system and the steps involved in it. In this topic , we will show the implementation of the proposed system in the code snippet format from the starting to the end .

Code snippet 1: Importing of Pandas and Numpy as well as Matplotlib library and displaying the dataset.

```
import numpy as np
import pandas as pd
train=pd.read_excel(r'E:\train_data.xlsx')
data=train
print(data)
```

Code snippet 2: Calculating the value count and displaying the graph using Seaborn Library.

```
data['Label'].value_counts()
import seaborn as sns
ax=sns.countplot(data.Label)
```

Code Snippet 3: Performing the Tokenization ,Stemming , Lemmetization and get the common words etc.

```
data['new_tweet']=data.Translated_text.str.replace('@','')
data.head()
data['new_tweet'] = data['new_tweet'].str.replace("[^a-zA-Z#]", " ")
data['new_tweet'] = data['new_tweet'].str.replace("#", "")
data.head()
from collections import Counter
all_words = []
for line in list(data['new_tweet']):
words = line.split()
for word in words:
all_words.append(word.lower())
a=Counter(all_words).most_common(10)
data['new_tweet'] = data['new_tweet'].apply(lambda x: x.split())
data.head()
from nltk.stem.snowball import SnowballStemmer
stemmer = SnowballStemmer("english")

data['new_tweet']= data['new_tweet'].apply(lambda x: [stemmer.stem(i) for i in x])
data.head()
```

Code Snippet 4: Using TF-IDF and Bag Of Words and splitting the dataset:

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(data["new_tweet"],
data["Label"], test_size = 0.2, random_state = 42)
print("training set :",x_train.shape,y_train.shape)
print("testing set :",x_test.shape,y_test.shape)
```

```
from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
count_vect = CountVectorizer(stop_words='english')
transformer = TfidfTransformer(norm='l2',sublinear_tf=True)
```

```
x_train_counts = count_vect.fit_transform(x_train)
x_train_tfidf = transformer.fit_transform(x_train_counts)

print(x_train_counts.shape)
print(x_train_tfidf.shape)
```

```
x_test_counts = count_vect.transform(x_test)
x_test_tfidf = transformer.transform(x_test_counts)

print(x_test_counts.shape)
print(x_test_tfidf.shape)
```

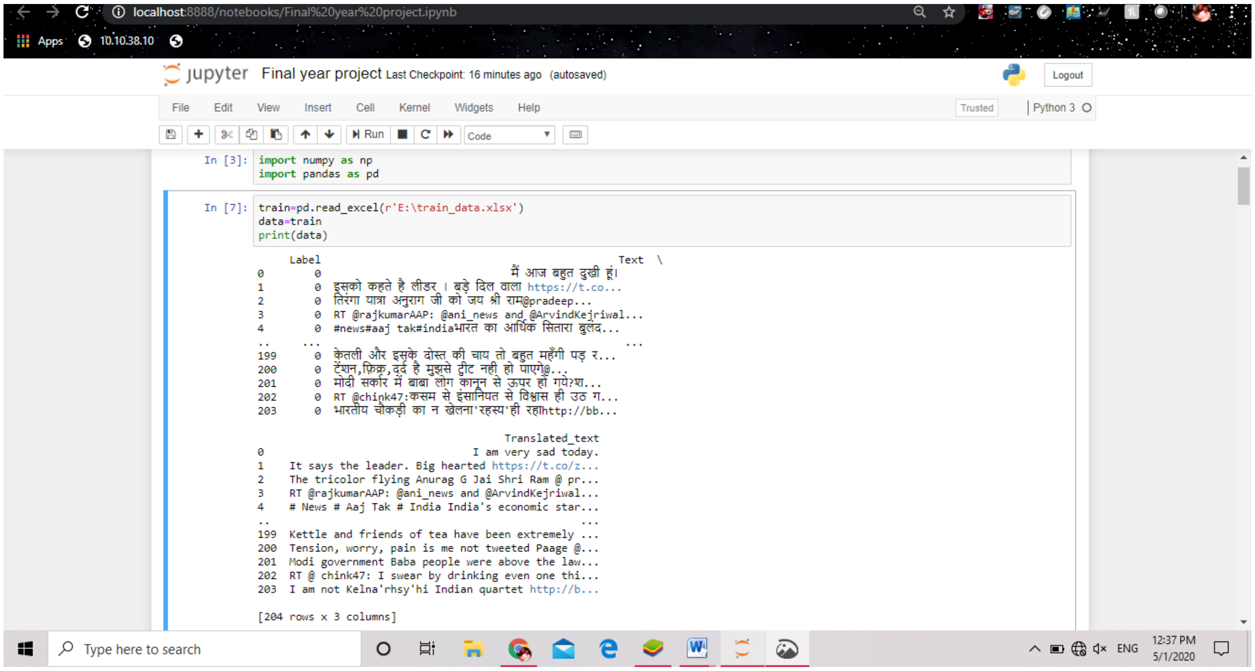
Code Snippet 5: Model Building with training and predicting the model:

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=200)
model.fit(x_train_tfidf,y_train)
predictions = model.predict(x_test_tfidf)
from sklearn.metrics import accuracy_score
accuracy_score(y_test,predictions)*100
from sklearn.metrics import confusion_matrix,f1_score
confusion_matrix(y_test,predictions)
f1_score(y_test,predictions)
```

OUTPUT/RESULT/SCREENSHOT

In this previous section , we showed the implementation of the project in the code snippet showing the working of the model. Now that we have completed the implementation we will show the output with the help of screenshots for each code snippet we showed in previous section and are mostly important.

Output of Code Snippet 1:



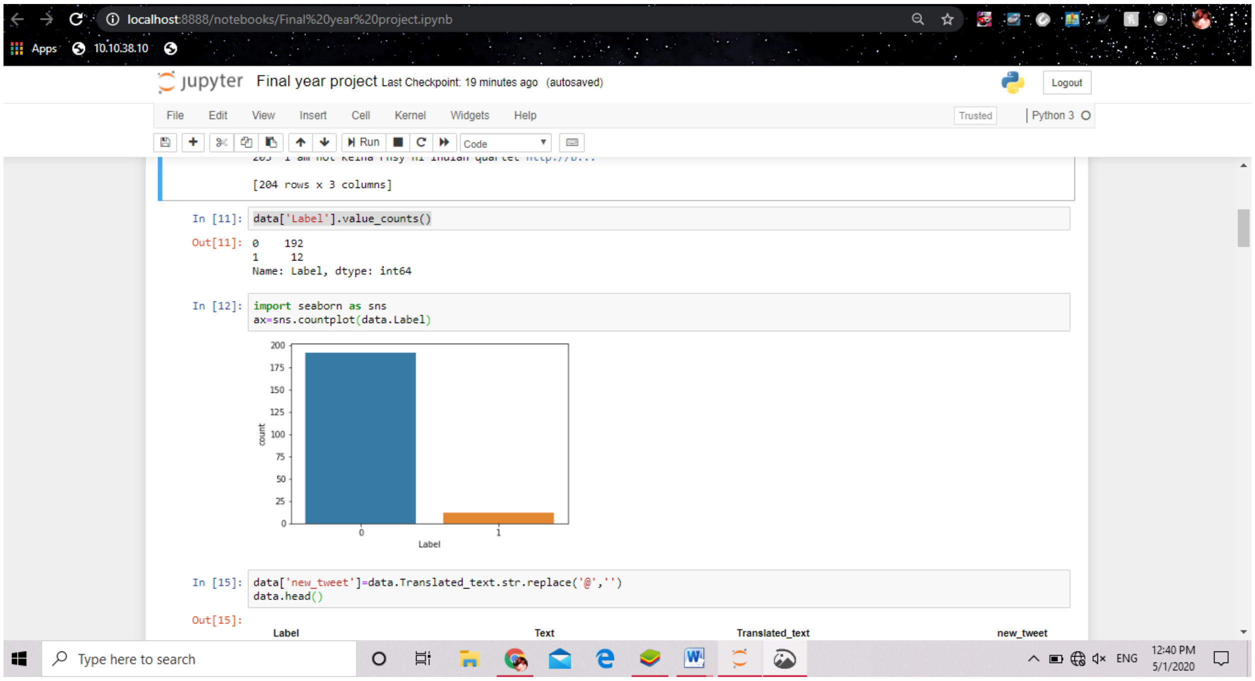
```
In [3]: import numpy as np
        import pandas as pd
```

```
In [7]: train=pd.read_excel(r'E:\train_data.xlsx')
        data=train
        print(data)
```

```
     Label                                               Text  \
0        0                                   मैं आज बहुत दुखी हूं।
1        0  इसको कहते है लीडर । बड़े दिल वाला https://t.co...
2        0  तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...
3        0  RT @rajkumarAAP: @ani_news and @ArvindKejriwal...
4        0  #news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...
..     ...                                                ...
199      0  केतली और इसके दोस्त की चाय तो बहुत महँगी पड़ र...
200      0  टेंशन,फ़िक्र,दर्द है मुझसे ट्वीट नही हो पाएगे@...
201      0  मोदी सर्कार में बाबा लोग कानून से ऊपर हो गये?श...
202      0  RT @chink47:कसम से इंसानियत से विश्वास ही उठ ग...
203      0  भारतीय चौकड़ी का न खेलना'रहस्य'ही रहाhttp://bb...

                                       Translated_text
0                                I am very sad today.
1    It says the leader. Big hearted https://t.co/z...
2    The tricolor flying Anurag G Jai Shri Ram @ pr...
3    RT @rajkumarAAP: @ani_news and @ArvindKejriwal...
4    # News # Aaj Tak # India India's economic star...
..                                                 ...
199  Kettle and friends of tea have been extremely ...
200  Tension, worry, pain is me not tweeted Paage @...
201  Modi government Baba people were above the law...
202  RT @ chink47: I swear by drinking even one thi...
203  I am not Kelna'rhsy'hi Indian quartet http://b...

[204 rows x 3 columns]
```

Figure 2: Output of Code Snippet 1

Output of Code Snippet 2:

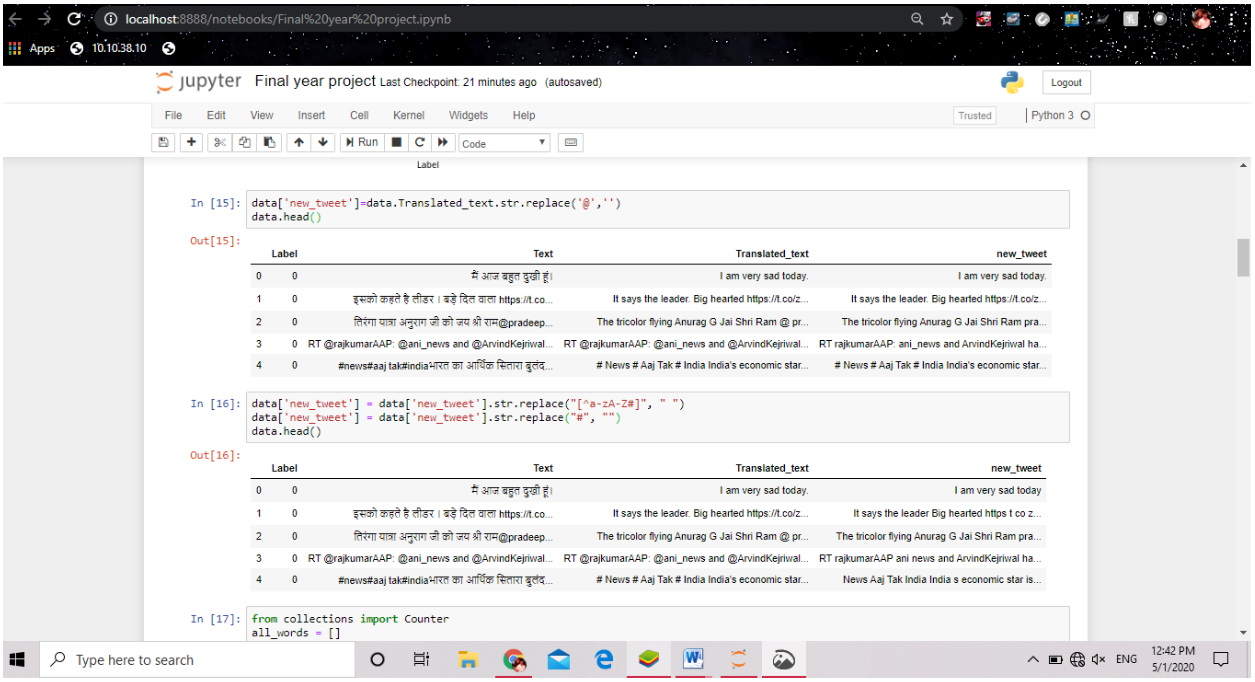


```
[204 rows x 3 columns]

In [11]: data['Label'].value_counts()

Out[11]: 0    192
         1     12
         Name: Label, dtype: int64

In [12]: import seaborn as sns
         ax=sns.countplot(data.Label)

In [15]: data['new_tweet']=data.Translated_text.str.replace('@','')
         data.head()

Out[15]:
```

	Label	Text	Translated_text	new_tweet

Figure 2: Output of Code Snippet 2

Output of Code Snippet 3:



```
In [15]: data['new_tweet']=data.Translated_text.str.replace('@','')
         data.head()

Out[15]:
```

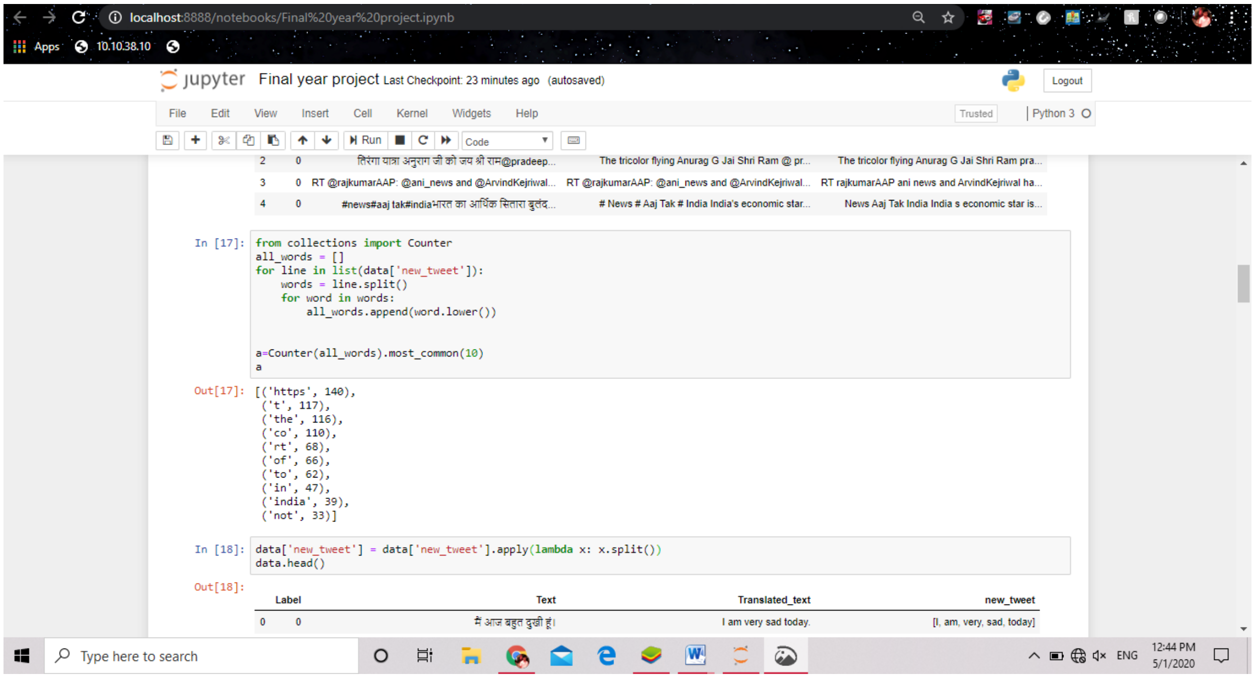
	Label	Text	Translated_text	new_tweet
0	0	मैं आज बहुत दुखी हूं।	I am very sad today.	I am very sad today.
1	0	इसको कहते है लीडर । बड़े दिल वाला https://t.co...	It says the leader. Big hearted https://t.co/z...	It says the leader. Big hearted https://t.co/z...
2	0	तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...	The tricolor flying Anurag G Jai Shri Ram @ pr...	The tricolor flying Anurag G Jai Shri Ram pra...
3	0	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT rajkumarAAP: ani_news and ArvindKejriwal ha...
4	0	#news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...	# News # Aaj Tak # India India's economic star...	# News # Aaj Tak # India India's economic star...

```
In [16]: data['new_tweet'] = data['new_tweet'].str.replace("[^a-zA-Z#]", " ")
         data['new_tweet'] = data['new_tweet'].str.replace("#", "")
         data.head()

Out[16]:
```

	Label	Text	Translated_text	new_tweet
0	0	मैं आज बहुत दुखी हूं।	I am very sad today.	I am very sad today
1	0	इसको कहते है लीडर । बड़े दिल वाला https://t.co...	It says the leader. Big hearted https://t.co/z...	It says the leader Big hearted https t co z...
2	0	तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...	The tricolor flying Anurag G Jai Shri Ram @ pr...	The tricolor flying Anurag G Jai Shri Ram pra...
3	0	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT rajkumarAAP ani news and ArvindKejriwal ha...
4	0	#news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...	# News # Aaj Tak # India India's economic star...	News Aaj Tak India India s economic star is...

```
In [17]: from collections import Counter
         all_words = []
```

Figure 3: Output of Code Snippet 3 – Replacing usernames and punctuations

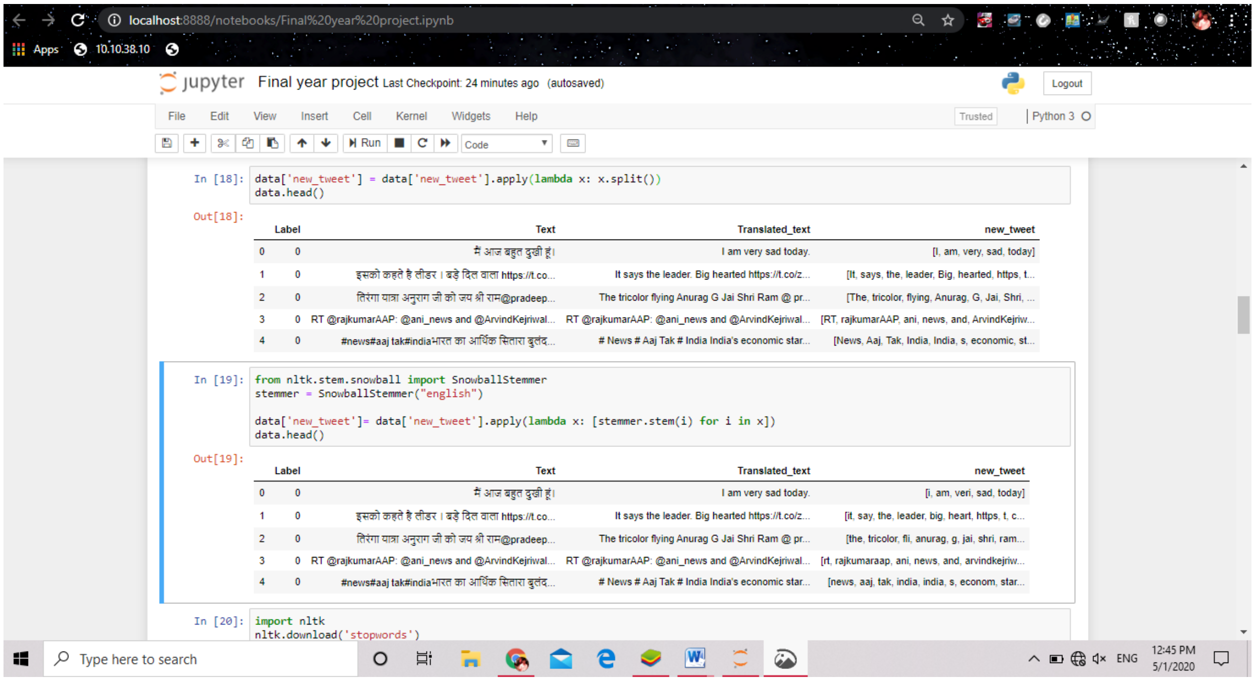

localhost:8888/notebooks/Final%20year%20project.ipynb
Apps
10.10.38.10
jupyter Final year project Last Checkpoint: 23 minutes ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
Type here to search
ENG
12:44 PM
5/1/2020

2	0	तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...	The tricolor flying Anurag G Jai Shri Ram @ pr...	The tricolor flying Anurag G Jai Shri Ram pra...
3	0	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT rajkumarAAP ani news and ArvindKejriwal ha...
4	0	#news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...	# News # Aaj Tak # India India's economic star...	News Aaj Tak India India s economic star is...

```
In [17]: from collections import Counter
         all_words = []
         for line in list(data['new_tweet']):
             words = line.split()
             for word in words:
                 all_words.append(word.lower())


         a=Counter(all_words).most_common(10)
         a

Out[17]: [('https', 140),
          ('t', 117),
          ('the', 116),
          ('co', 110),
          ('rt', 68),
          ('of', 66),
          ('to', 62),
          ('in', 47),
          ('india', 39),
          ('not', 33)]

In [18]: data['new_tweet'] = data['new_tweet'].apply(lambda x: x.split())
         data.head()

Out[18]:
```

	Label	Text	Translated_text	new_tweet
0	0	मैं आज बहुत दुखी हूं।	I am very sad today.	[I, am, very, sad, today]

Figure 4: Output of Code Snippet 3- using common words and tokenizing


localhost:8888/notebooks/Final%20year%20project.ipynb
Apps
10.10.38.10
jupyter Final year project Last Checkpoint: 24 minutes ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
Type here to search
ENG
12:45 PM
5/1/2020

```
In [18]: data['new_tweet'] = data['new_tweet'].apply(lambda x: x.split())
         data.head()

Out[18]:
```

	Label	Text	Translated_text	new_tweet
0	0	मैं आज बहुत दुखी हूं।	I am very sad today.	[I, am, very, sad, today]
1	0	इसको कहते है लीडर । बड़े दिल वाला https://t.co...	It says the leader. Big hearted https://t.co/z...	[It, says, the, leader, Big, hearted, https, t...
2	0	तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...	The tricolor flying Anurag G Jai Shri Ram @ pr...	[The, tricolor, flying, Anurag, G, Jai, Shri, ...
3	0	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	[RT, rajkumarAAP, ani, news, and, ArvindKejriw...
4	0	#news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...	# News # Aaj Tak # India India's economic star...	[News, Aaj, Tak, India, India, s, economic, st...

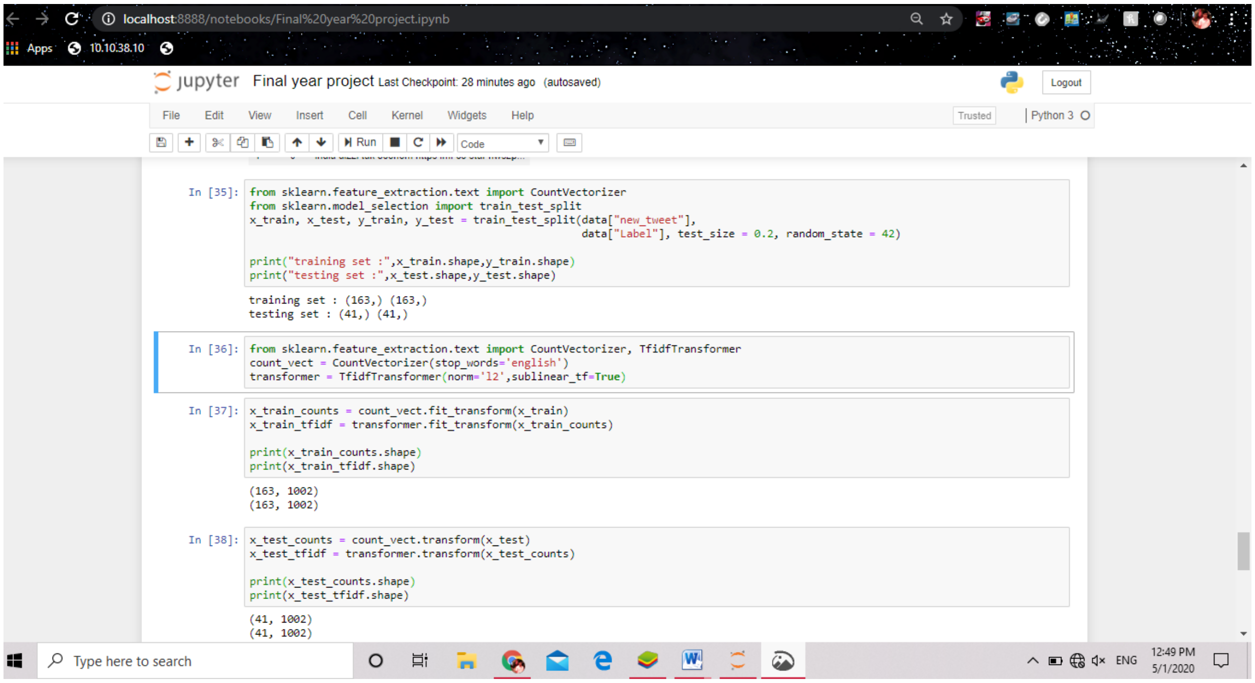
```
In [19]: from nltk.stem.snowball import SnowballStemmer
         stemmer = SnowballStemmer("english")

         data['new_tweet']= data['new_tweet'].apply(lambda x: [stemmer.stem(i) for i in x])
         data.head()

Out[19]:
```

	Label	Text	Translated_text	new_tweet
0	0	मैं आज बहुत दुखी हूं।	I am very sad today.	[i, am, veri, sad, today]
1	0	इसको कहते है लीडर । बड़े दिल वाला https://t.co...	It says the leader. Big hearted https://t.co/z...	[it, say, the, leader, big, heart, https, t, c...
2	0	तिरंगा यात्रा अनुराग जी को जय श्री राम@pradeep...	The tricolor flying Anurag G Jai Shri Ram @ pr...	[the, tricolor, fli, anurag, g, jai, shri, ram...
3	0	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	RT @rajkumarAAP: @ani_news and @ArvindKejriwal...	[rt, rajkumaraap, ani, news, and, arvindkejriw...
4	0	#news#aaj tak#indiaभारत का आर्थिक सितारा बुलंद...	# News # Aaj Tak # India India's economic star...	[news, aaj, tak, india, india, s, econom, star...

```
In [20]: import nltk
         nltk.download('stopwords')
```

Figure 5: Output of Code Snippet 3-using stemming and lemmetisation

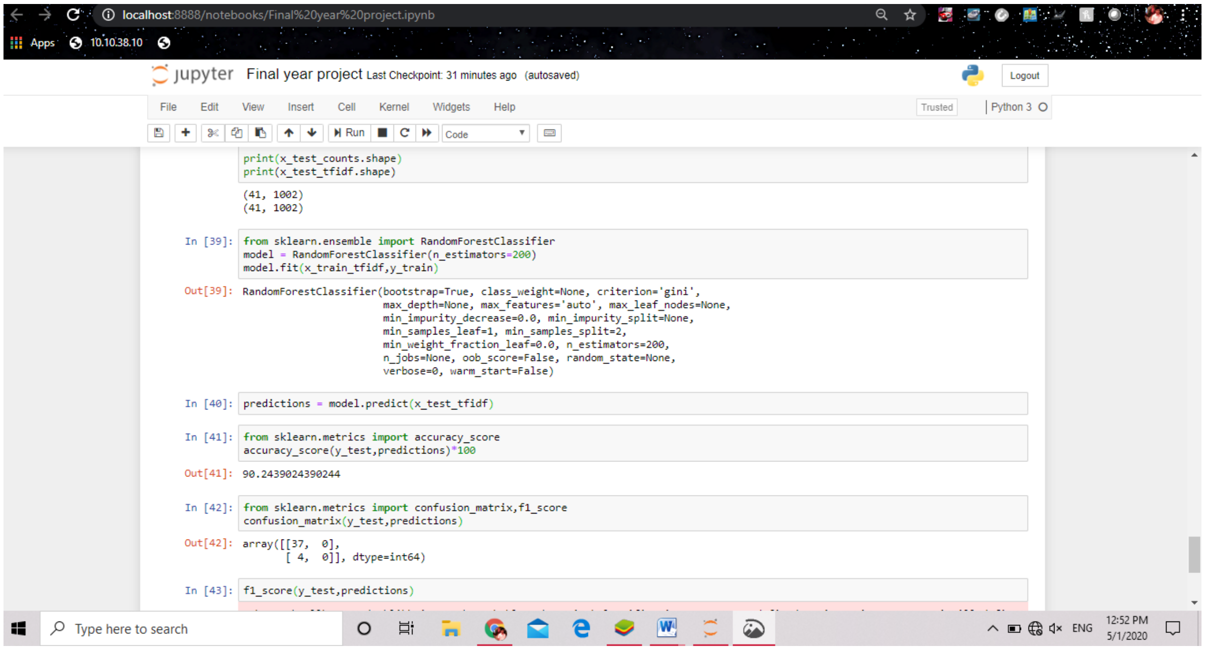
Output of Code Snippet 4:



```
In [35]: from sklearn.feature_extraction.text import CountVectorizer
         from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(data["new_tweet"],
                                                             data["Label"], test_size = 0.2, random_state = 42)

         print("training set :",x_train.shape,y_train.shape)
         print("testing set :",x_test.shape,y_test.shape)

         training set : (163,) (163,)
         testing set : (41,) (41,)

In [36]: from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
         count_vect = CountVectorizer(stop_words='english')
         transformer = TfidfTransformer(norm='l2',sublinear_tf=True)

In [37]: x_train_counts = count_vect.fit_transform(x_train)
         x_train_tfidf = transformer.fit_transform(x_train_counts)

         print(x_train_counts.shape)
         print(x_train_tfidf.shape)

         (163, 1002)
         (163, 1002)

In [38]: x_test_counts = count_vect.transform(x_test)
         x_test_tfidf = transformer.transform(x_test_counts)

         print(x_test_counts.shape)
         print(x_test_tfidf.shape)

         (41, 1002)
         (41, 1002)
```

Figure 6: Using Bag of Words and TF-IDF count Vectorizer and splitting the train and test dataset.

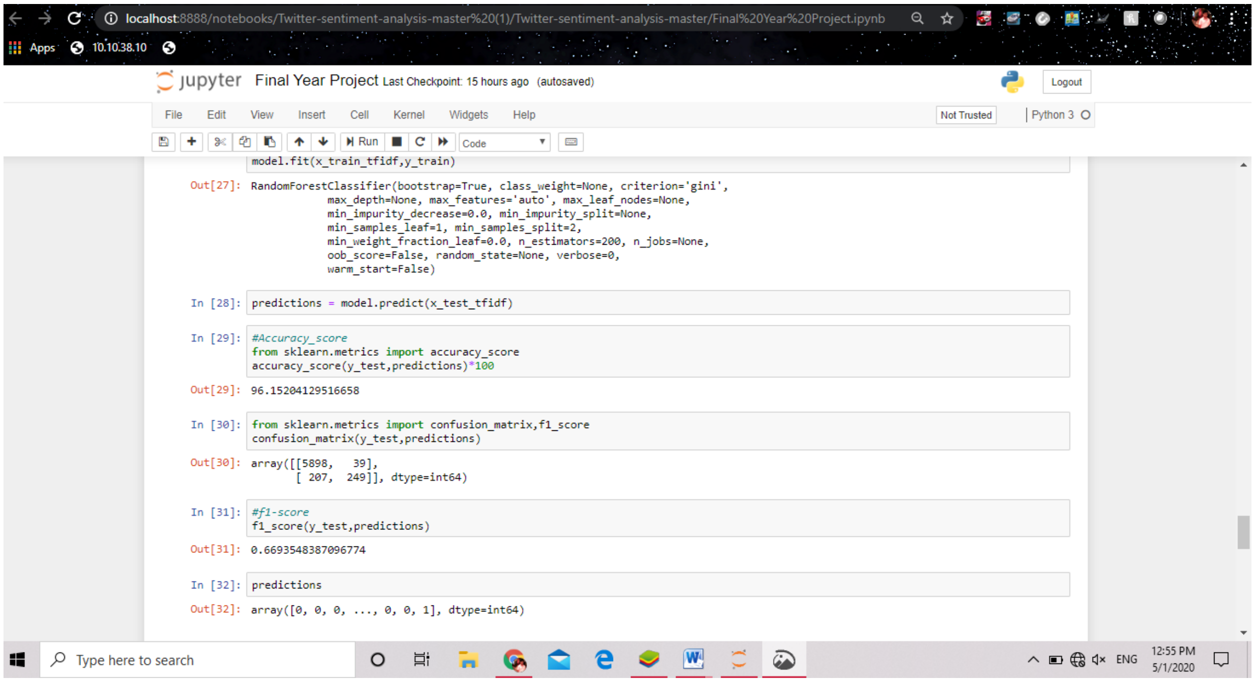
Output of Code Snippet 5:



```
         print(x_test_counts.shape)
         print(x_test_tfidf.shape)

         (41, 1002)
         (41, 1002)

In [39]: from sklearn.ensemble import RandomForestClassifier
         model = RandomForestClassifier(n_estimators=200)
         model.fit(x_train_tfidf,y_train)

Out[39]: RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                                max_depth=None, max_features='auto', max_leaf_nodes=None,
                                min_impurity_decrease=0.0, min_impurity_split=None,
                                min_samples_leaf=1, min_samples_split=2,
                                min_weight_fraction_leaf=0.0, n_estimators=200,
                                n_jobs=None, oob_score=False, random_state=None,
                                verbose=0, warm_start=False)

In [40]: predictions = model.predict(x_test_tfidf)

In [41]: from sklearn.metrics import accuracy_score
         accuracy_score(y_test,predictions)*100

Out[41]: 90.2439024390244

In [42]: from sklearn.metrics import confusion_matrix,f1_score
         confusion_matrix(y_test,predictions)

Out[42]: array([[37,  0],
                [ 4,  0]], dtype=int64)

In [43]: f1_score(y_test,predictions)
```

Figure 7: Importing of the Random Forest and training and testing the dataset.

Output of Code Snippet 5:


localhost:8888/notebooks/Twitter-sentiment-analysis-master%20(1)/Twitter-sentiment-analysis-master/Final%20Year%20Project.ipynb
Apps
10.10.38.10
jupyter
Final Year Project Last Checkpoint: 15 hours ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
Type here to search
ENG
12:55 PM
5/1/2020

```
          model.fit(x_train_tfidf,y_train)

Out[27]:  RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                      max_depth=None, max_features='auto', max_leaf_nodes=None,
                      min_impurity_decrease=0.0, min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, n_estimators=200, n_jobs=None,
                      oob_score=False, random_state=None, verbose=0,
                      warm_start=False)

In [28]:  predictions = model.predict(x_test_tfidf)

In [29]:  #Accuracy_score
          from sklearn.metrics import accuracy_score
          accuracy_score(y_test,predictions)*100

Out[29]:  96.15204129516658

In [30]:  from sklearn.metrics import confusion_matrix,f1_score
          confusion_matrix(y_test,predictions)

Out[30]:  array([[5898,   39],
                 [ 207,  249]], dtype=int64)

In [31]:  #f1-score
          f1_score(y_test,predictions)

Out[31]:  0.6693548387096774

In [32]:  predictions

Out[32]:  array([0, 0, 0, ..., 0, 0, 1], dtype=int64)
```

Figure 8: Calculating result using Accuracy score and f1 score.

CONCLUSION/FUTURE ENHANCEMENTS

The implementation of the project concludes that the f1 score is achieved by 0.66354 by using Random Forest Algorithm for building and testing the model. This proves that the Hindi Tweets dataset implemented by Random Forest has a f1 score of 0.66354 and accuracy score of 90.24.

The future enhancements of this Sentiment Analysis is that automatic sentiment analysis has a fair way to go before it can replace human coding of sentiment - though even human coding will have problems, as my idea of negative or somewhat negative may well be different from yours. More and more, I am convinced that machine learning techniques and sophisticated text analytics algorithms will be needed to improve the accuracy of automatic sentiment analysis. Having said that, I believe that sentiment analysis will only increase in importance as more and more people use online channels to communicate, both directly and indirectly, with corporations.

REFERENCES

Journal/Articles

[1]. Kalaivani A , Thenmozhi D did a survey on Deep Learning where they presented paper on Sentiment Analysis using Deep Learning Techniques doing their survey published at IJRTE in April 2019.

[2]. Apoorv Agarwal , Boyi Xie , Ilia Vovsha , Owen Rambow , Rebecca Passonneau ,"Sentiment Analysis on Twitter Data.

[3]. Duyu Tang , Furu Wei(2014) developed a Deep Learning System , called "Coooll" which is a deep learning model that builds sentiment classifier from Tweets and manually annotated sentiment polarity.

[4]. Varsha Sahayak , Vijaya Shete (Jan 2015) , "sentiment analysis on Twitter Data."

[5]. Dimitris Effrosynidis (21st Conference on Theory and Practice of Digital Libraries).

[6]. Adyan Marendra Ramdhani(2017) , "Twitter Sentiment Analysis using Deep Learning."

[7] Aanusha Ghosh and Indranil Dutta , "Real Time Sentiment Analysis using Hindi Tweets" .

[8] Prateek Garg (2016) , " Sentiment Analysis on Twitter Data using NLTK ".

Book Journal- Progress in Advanced Computing and Intelligent Engineering pp 573-580

[9]. Taranpreet Singh Ruprah , Nitin Trivedi " Sentiment Analysis on Twitter Data using Data Mining Techniques , Part of the Advances in Intelligent Systems and Computing book series (AISC, volume 713) , December 2018.

[10] A. Sanghvi , Darshin K. Shah , Artika Singh , "Sentiment Analysis of Tweets using Supervised Learning Techiques , Part of the Advances in Intelligent Systems and Computing book series (AISC , volume 1045) , November 2019.

[11] Sai Srihitha Yadlapalli , R. Rakesh Reddy , T. Sasikala **, "**Advanced Twitter Sentiment Analysis Using Supervised Techniques and Minimalistic Features" Part of Advances in Intelligent Systems and Computing book series (AISC , volume 1097) , March 2020.

[12] Sandip Pali , Soumadip Ghosh , "Real Time Sentiment Analysis" , Copyright of International Journal of Synthetic Emotions(IJSE) 2020.

[13] Chaitanya Bhagat , Deepak Mane , "Text Categorisation using Sentiment Analysis" , Part of the Proceedings of International Conference on Computational Science and Application , Conference-January 2020.

[14] Shihab Elbagir , Yin Yang , "Sentiment Analysis of Twitter Data using Python NLTK VADER'S Python Analyzer" , IAENG Transactions on Engineering Sciences ,2020.