Real-Time Sentiment Analysis Towards Machine Learning

Swati Sharma, Dr. Mamta Bansal

Abstract: Machine Learning is one of the apogees growing data science having tremendously large domain of applications. It is a subfield of data science; having breathtaking approaches in the recent years and will surely grow with a rapid rate. According to current scenario, the use and demand of social media is increasing tremendously. People are relying more on internet, social media and thus the size of the data is becoming huge day by day. As the size of data is extremely huge, so there is a frightening rate to analyze the data and to extricate the useful data various technologies like big data, cloud computing, machine learning, deep learning, data science are coming up in the market. In this paper, we are analyzing tweets. From twitter, we have collected a large number of tweets, tokenization is performed using anaconda, an integrated development environment of python and then training is done on the tokenized tweets to identify the useful data. Spyder is yet another powerful technical environment of python. It provides advanced features of data analysis as well as excellent visualization.

Index Terms: Anaconda, Data Science, Machine Learning, Natural Language Processing, Snowballstemmer, Spacy, Spyder, Tweepy.

1. INTRODUCTION

SENTIMENT analysis, also refer as opinion mining is a family of data mining which measures the propensity of an individual's feeling using data mining, data science, data analysis, machine learning etc. [5] Sentiment analysis focuses to measure the individual behavior with respect to some topic or event whether he or she is feeling positive, negative or neutral. To distinguish people's opinion various supervised learning and unsupervised learning approaches are used. [1]Text analytics is the method of obtaining polished information from data. [4]The polished information is obtained from the training of patterns using methods like statistical pattern learning. The procedure of text mining basically includes the classification of input data, obtaining pattern from the classified data and finally evaluation and the analysis of the obtained output. It is similar to data mining. For data analysis, we have used python; a high-level programming language, including various programming prototypes. Anaconda is an open source integrated development environment for python. It is helpful in evolving, training deep learning and machine learning models. [11]It is helpful in examining data with versatility. Anaconda is a cloud where various packages like numpy, mathplotlib, scipy, spacy etc. are shared. Machine Learning is the technical study of data structures, algorithms and statistical models which systems use to execute the work without the help of explicit conditions. It is interrelated to computational statistics. [3]Machine learning can be accomplished by supervised learning, semi-supervised learning and unsupervised learning. Data science is a field which uses scientific methods, algorithms, procedures to extricate knowledge from huge amount of structured as well as unstructured data. [2]It involves formation of data insight as well as data products.

2 METHODOLOGY

2.1 Filtration of Raw Data

Detailed Raw data, also known as source data is a type of data that has not yet processed. There is a minor difference between data and information.

[10]Data are the raw facts or materials which has its implicit meaning whereas the processed form of data is known as information. The data is being collected from twitter. An application is being created on twitter, on its application.The following Consumer key, consumer_secret, access_token, access_token_secret key is generated as follows:

```
consumer_key = "Sh4rtCV78EummAG1yWaQaXmyw"
consumer_secret =
"h6BTyRKOKPoI7ucXiZRNmiJuFTM4wvsu8adIU5rt5qVy6N6T
fX"
access_token = "1046769984737665029-
y7HJ9s9wBEyDkJKHc5ik3CCyX083Q3"
access_token_secret =
"j51htadkbtPBkuiL46ZlWp1fB7tlXREia32UarRH5WKAv" by
```

Handshake authorization is being generated between consumer and its account as follows:

```
auth = tweepy.OAuthHandler(consumer_key,
consumer_secret)
auth.set_access_token(access_token, access_token_secret )
api = tweepy.API(auth,wait_on_rate_limit=True )
```

To use its services on anaconda, tweepy is being installed on anaconda prompt using

```
Conda install –c conda-forgetweepy

# Open/Create a file to append data

csvFile = open('tryyy.csv', 'a')
#Use csv Writer
csvWri   ter = csv.writer(csvFile)
csvWriter.writerow(["date", "twee"])
for tweet in
tweepy.Cursor(api.search,q="#election",count=4000,
lang="en",
date_since="2019-01-01").items():
        print (tweet.created_at, tweet.text)
csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-
8')])
```


- *Swati Sharma is currently pursuing ph.d from Shobhit University, Meerut, India, PH-9808150268. E-mail: swati.sharma.it@miet.ac.in*
- *Dr. Mamta Bansal, Shobhit University, Meerut, India, PH-6395435034. E-mail: mamta.links@gmail.com*

```
p=[]
d=pd.read_csv("sample tweets.csv")
for i in d.twee:
p.append(i)
```

Thus, a sample of random tweets are collected from twitter in English language.

2.2 Tokenization of Filtered Data

Tokenization is a process of breaking up of task into parts also known as tokens and removing some features such as stop words. The loose words are known as tokens but it is beneficiary to distinguish among tokens.

For doing tokenization using anaconda, install natural language toolkit as follows:

Conda install –c anaconda nltk

There is a stemmer named snowballstemmer which needs to be imported from NLTK, it removes morphological post from tokens.

There is a package named spacy, it is a open source library used for natural language processing in anaconda. It is basically used for production and dealing with huge amount of text. [9]It is desirable to preprocess the text for deep learning.

Import Spacy as follows:

Conda install –c conda-forge spacy

The data is being tokenized after removing stop words, cue words, punctuations etc., which leads in the reduction in the size of data.[7] A set of tokenized tweets will be obtained.

```
e=[]
fi=pd.read_csv("tweetsafterrr.csv")
fi.rename(columns={"tweets after tokenization":"A"})
fori in fi.A:
                    print(i)
                        for word in i:
                            print(word)
e.append(s_stemmer.stem(word))
```

For papers accepted for publication, it is essential that the electronic version of the manuscript and artwork match the hardcopy exactly! The quality and accuracy of the content of the electronic material submitted is crucial since the content is not recreated, but rather converted into the final published version.

2.3 Vectorization of Tokenized Tweets

Vectorization is a very important feature which is used to increase the execution speed of python code without using loop and thus the efficiency of the code increases. [8]There are number of ways to do vectorization such as scalar product, outer product, element wise multiplication, dot product etc.

In spyder, integrated development environment of python, pipeline is used as follows:

```
fromsklearn.pipeline import Pipeline
fromsklearn.feature_extraction.text  import TfidfVectorizer
fromsklearn.svm import LinearSVC
```

2.4 Training of Vectorized Tweets

After vectorization, the method of training using machine learning comes. The data which has to be trained must possess right output referred to as target attribute. [6]The learning procedure captures pattern in the trained data which binds the input data attributes to the output data attributes. There is a classifier in machine learning known as support vector classifier; it is a supervised learning model which assesses data used for regression analysis and classification. It can also be used as a non-linear classifier:

To import linear support vector classifier, use:

```
text_clf_lsvc = Pipeline([('tfidf', TfidfVectorizer()),
            ('clf', LinearSVC()),])
text_clf_lsvc.fit(X_train, y_train)
predictions = text_clf_lsvc.predict(X_test)
from sklearn.externals import joblib
filename = 'finalizedl.sav'
joblib.dump(text_clf_lsvc, filename)
loaded_model = joblib.load(filename)
result = loaded_model.score(X_test, y_test)
```

3 RESULTS AND CONCLUSION

In this paper, we have worked upon integrated development environment of python known as anaconda. We have used spyder, a free IDE, an app of anaconda. It provides scientific python development environment. The data is collected from twitter. After collecting data, tokenization is performed which leads in the reduction in the size of dataset. Then vectorization done by which the efficiency of the python code is increased and finally training is done on the basis of the vectorized dataset using support vector class. The figure1 shows the precision comparison between trained and random tweets, how precise the random and trained tweets are to each other. The figure 2 shows the recall comparison among trained and random tweets, the recall factor is much more accurate of trained tweets in comparison to random tweets. The figure 3 shows the f1-score comparison between random and trained tweets, it has been seen that trained tweets have higher score than random tweets.

Fig. 1 *— Precision Comparison among Random and Trained Tweets*

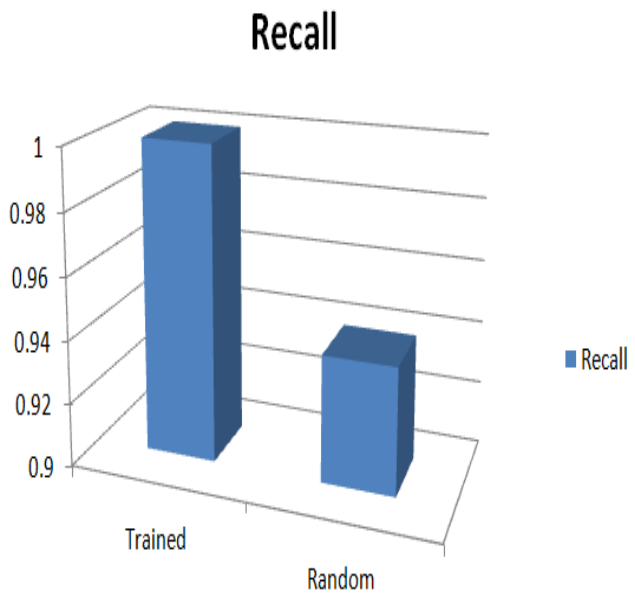

Fig. 2 *— Recall Comparison among Random and Trained Tweets*

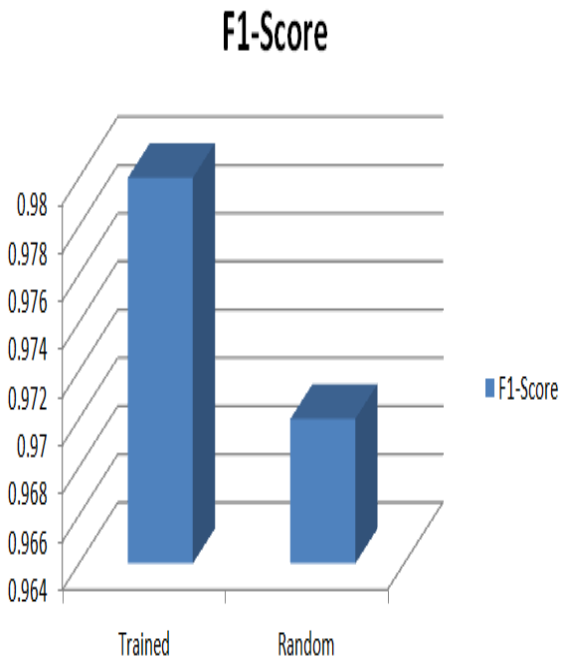

Fig. 3 *— F1-Score Comparison among Random and Trained Tweets*

```
In [1]: runfile('C:/Users/swati sharma/Downloads/training.py', wdir='C:/Users/swati
sharma/Downloads')
[[3883    0]
 [ 122 1881]]
              precision    recall  f1-score   support

    election       0.96      1.00      0.98      3883
      random       1.00      0.94      0.97      2003

    accuracy                           0.98      5886
   macro avg       0.98      0.97      0.97      5886
weighted avg       0.98      0.98      0.98      5886
```

Figure 4 *— Comparison of Random and Trained Tweets*

The figure 4 represents a comparison between trained and random tweets, their accuracy, macro average and weighted average are compared.

REFERENCES

[1] Borele P and Borilar A, “An approach to sentiment Analysis using Artificial Neural Network with comparative Analysis of Different Techniques”. IOSR Volume 18: 64-69, 2016.

[2] Pan S, Ni X and Sun J, “Cross-domain sentiment classification via spectral feature alignment”. Proceedings of the 19th international conference on World Wide Web ACM: 751-760, 2010.

[3] Liu S, Li F, Cheng X, and Shen H, “Adaptive curtaining SVM for sentiment classification on tweets” In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management: 2079-2088, 2013.

[4] Tajinder S and Kumari M, “Role of Text Pre-Processing in Twitter Sentiment Analysis”, Procedia Computer Science, 2016.

[5] Vaitheewaran G and Arockiam L, “Combining Lexicon and Machine Learning Method to enhance the accuracy of Sentiment Analysis on Big Data” International Journal of Computer Science and Information Technology Volume 6: 306-311, 2016.

[6] Leszek Z, “The sentiment analysis as a tool of business analytics in contemporary organizations” 234-241, 2016.

[7] Kumari U, Soni D and Sharma A, “A Cognitive study of Sentiment Analysis Techniques and Tools: A Survey” International Journal of Computer Science and Technology Volume 8: 58-62, 2017.

[8] Tao Chen, Ruifeng Xu, Yulen He and Xuan Wang, “Improving sentiment analysis via sentiment type classification using BiLSTM-CRF and CNN” Expert Systems with ApplicationsVolume 72: 221-230, 2017.

[9] Mandava G and Rajeswara R, “Sentiment Analysis on social media using R programming”, International Journal of Engineering and Technology Volume12: 148-153, 2018.

[10] Vyas V and Uma V, “An extensive study of Sentiment analysis tools and binary classification of tweets using Rapid Miner”, Volume 11: 247-254, 2018.

[11] Mavi Vand Tyagi N, “Hadoop’s Second Generation – YARN”, International Journal of Contemporary Research in Engineering & Technology Volume7: 2250-0510, 2017.