Journal of Engineering and Applied Sciences 13 (Special Issue 5): 4600-4603, 2018
ISSN: 1816-949X

A Twitter based Sentimental Approach on India's Largest 4G Network "JIO"

[1]V. Reshma, [1]Sharanya Anand and [2]Maya L. Pai
[1]Department of Mathematics,
[2]Department of Computer Science, School of Arts and Science,
Amrita University Kochi, Kerala, India

Abstract: Sentimental analysis states the use of Natural Language Processing (NLP). It is used to track the emotions or sentiments of the people based on a particular product or topic. The micro blogging websites like Facebook and Twitter plays an inevitable role in tracing these emotions. This study mainly focuses on fitting a model based on classification of Tweets as positive and negative using different filters namely Naive Bayesian and filtered classifiers using machine learning tool and to prove how effective and accurate the machine learning tool can be used in data mining to predict Tweets as positive and negative. It is observed that Naive Bayesian filter gives better results than filtered classifier for test data and further can be employed as a tool to study the opinions of people.

Key words: Textblob, Tweepy, API, Naive Bayesian, filtered classifier, opinions

INTRODUCTION

Sentiment analysis or opinion mining is the process of defining an emotional tendency used to develop and analyze public's opinions, sentiments, evaluation, appraisals, attributes and emotions expressed within an online platform. It also deals with the study of thoughts from text. The sense of opinion itself is still very wide. Sentiment analysis mainly focuses on views or estimation which imply positive or negative sentiments. The study is also effective to analyze the opinions of the end users on a particular product or services which in turn can be productively used for the improvement of the quality of the goods or services.

Literature review: Davidov *et al*. (2010) has proposed a supervised sentiment classification based on Twitter data and successfully identifies opinions of untagged sentiments. Kalaivani and Shunmuganathan (2013) suggested a model where sentiment classification techniques were applied to movie reviews. A comparison was done between the three supervised machine learning approaches, SVM, Naive approaches, SVM, Naive Bayes and KNN. SVM showed the highest accuracy with 80%. Shruti *et al*. (2009) evaluated the impact of Tweets containing emoticons to the classifying process using machine learning tools and experimental results outperformed Naive Bayes algorithms. Nasukawa and Nagano (2001) defined a technique to extract a set of sentiment units from sentences which is the key element of sentiment analysis. Sentiment unit is a tuple of sentiments. It mainly focuses on individual sentiment expression which was not only meant to meet the overall goodness of an object but also the breakdown of opinions. Agarwal *et al*. (2011) examined Twitter analysis on sentiment data. They explored the use of kernel tree to obviate the need for engineers. Pang *et al*. (2002) focused on methods that seek to address the new challenges raised by sentiment analysis as compared to those with traditional fact based analysis. Efthymios Kouloumpis investigated the utility of linguistic features for detecting the sentiment of Twitter messages. Vinodhini and Chandrasekaran (2012) presented a survey covering the techniques and methods in sentiment analysis and challenges appearing in the field. Mika Viking created taxonomy of research topics with text mining and qualitative coding and a future for sentiment analysis ensured in detecting fake news. Jandail (2014) demonstrates the possibility of the idea through clustering and classifying opinion mining experiment on analysis of blog posts on recent product policies and service reviews. Jagtap and Karishma Pawar focused on sentiment classification in sentences and analyzed a solution for sentiment analysis using polarity. Mohini Chaudhary and Sharvari Govilkar focused on different machine learning techniques used for sentiment analysis and compared the different methods on the basis of accuracy.

Data: Twitter is a micro blogging website which allows users to publish short messages called as Tweets. The data used for this study are in the form of Tweets. Tweets can be accessed from Twitter by creating an application in Twitter.

Corresponding Author: V. Reshma, Department of Mathematics, School of Arts and Science, Amrita University Kochi, Kerala, India

To access the Twitter data programmatically, it requires to create an application that interacts with the Twitter account using Twitter Application Programming Interface (API). The first step is to log in to Twitter and register a new application. The registration of this application will take place in the Twitter website (http://apps.twitter.com). Then a name and a description for the application is selected for this study. It will create a consumer key and a consumer secret. These are application settings that should be kept confidential. From the configuration page of this application an access token and an access token secret is obtained. Similar to the consumer keys these strings must also be kept private. They provide the application access to Twitter on behalf of this account. Here, the default permissions are read only.

Python for sentiment analysis: Python is the programming language used for this study. It is a high level programming language which allows users to express concepts with fewer codes. Python is used to access the Twitter API, manipulate the data and save the output. Python is an interpreted language which allows programmers to use various programming styles to generate simple or complex programs in order to get faster results and to inscribe the codes similar to human language. The library Tweepy and TextBlob should be necessarily installed, so that, python library can access the Tweets in the form of texts. To interact with Twitter service, install Tweepy using pip command in python. In order to authorize this application to access Twitter, it requires to use the OAuth interface. Here we use Tweepy library to access the Twitter API and the Text blob library to perform sentiment analysis on each Tweet. From this we infer whether the sentiments of Tweets are positive, negative or neutral.

Tweepy: Tweepy is a library which is used to access the Twitter API with Python. The following command is used to install Tweepy with Python.

```
pip installation tweepy == 3.3.0
```

TextBlob: TextBlob is a python library used for the processing of textual data. It provides a simple API for diving into common natural language processing tasks such as classification, sentiment analysis, translation, parts of speech tagging and so on.

OAuth authentication: OAuth is an open standard interface for authorization, generally used as a path for internet handlers to support websites or applications to access their data on other websites without sharing the passwords. This authorization is utilized by certain Hyper Text Transfer Protocol (HTTP) services such as Facebook, Twitter, etc. by permitting the entities to share the information about their accounts with the third party applications or websites. OAuth interface helps in the authentication with Twitter API and python is used to retrieve these Tweets. After running the python code, it saves each Tweet to a CSV file with an associated label. The label should be either 'positive', 'negative'. The sentiment polarity thresholds are defined with the python script. The data contains different factors like username, location, Tweets, polarity and classification viz. positive, negative and neutral. From these attributes Tweets and classification (positive/negative) are considered for this study. The other attributes like username, location, polarity and Tweets classified as neutral are less significant and hence not considered in this study.

MATERIALS AND METHODS

The data received with the help of the Twitter application is then preprocessed. Data preprocessing is a vital step in the analysis which converts the raw data in to an understandable format and to formulate the same for another processing procedure up to high light all of the above researcher and affiliation.

Figure 1 depicts the flow chart that shows the step by step process of data processing. The preprocessed data is divided into two subsets. Training data and test data. The 70% of the whole data were chosen for training set and the rest were used for testing. Classification of training data set was done in machine learning tool, Weka using different filters. Initially, a classification model was fitted utilizing the training data by making use of various filters specifically Naive Bayesian and filtered classifier.

Naive Bayesian: The Bayesian classification designates a supervised learning method and a statistical method for classification. Bayesian filter helps in solving analytic and predictive problems. It is a family of probabilistic classifiers based of applying Bayes theorem. The basic principle behind it is all the attributes under consideration are independent of each other.

Filtered classifier: Filtered classifier is a supervised learning method which generally uses to choose attributes. It converts all nominal values to binary numeric values.

The classification model is fitted for the training set using these two filters. This classified model is then used for prediction. The accuracy is checked using the formula:

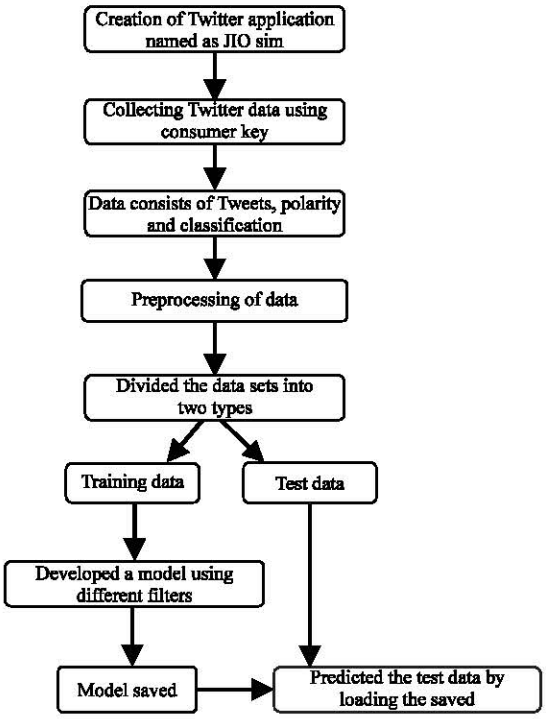

Fig. 1: Flow chart depicting the procedure

Table 1: Confusion matrix

Variables	Predicted (positive)	Predicted (negative)
Actual (positive)	TP	FN
Actual (negative)	FP	TN

$$\text{Accuracy} = \frac{TP + TN}{TP+TN+FP+PN}$$

Where:

TP = The positive output out where the actual input is also positive

TN = The true negatives where the actual input is also negative

FP = The positive output where the actual input is negative

FN = The negative output where the actual input is positive

Confusion matrix: Confusion matrix is a table used to define the performance of the classification model or classifier. It gives the details related to the actual and predicted value of the data in the classifier. It gives a relationship between the actual and predicted values. Table 1 shows the model of a confusion matrix were each of those values together contributes in finding the accuracy using different models.

RESULTS AND DISCUSSION

Using naive bayes filter: Table 2-4 depicts the detailed result analysis using Naive Bayesian filter. The accuracy for training data and test data of this filter resulted with 93.59 and 67.46%, respectively. Table 3 denotes the confusion matrix corresponding to this filter.

Table 2: Detailed analysis results in weka for Naive Bayesian filter

Terms	Values
Correctly classified instances	380
Incorrectly classified instances	26
Kappa statistic	0.8718
Mean absolute error	0.1141
Root mean squared error	0.2251
Relative absolute error	22.8546%
Root relative squared error	45.0423%

Table 3: Detailed accuracy by class in Naive Bayesian

Variables	TP rate	FP rate	Precision	Recall	ROC curve
Positive	0.934	0.062	0.934	0.934	0.981
Negative	0.938	0.066	0.938	0.938	0.981
Weighted average	0.936	0.064	0.936	0.936	0.981

Table 4: Confusion matrix for Naive Bayesian filter

Classifier	Filtered classifier	Naive Bayes
Accuracy of training data	87.89	93.59
Accuracy of predicted test data	56.62	67.46

Table 5: Detailed analysis results in weka for filtered classifier

Terms	Values
Correctly classified instances	354
Incorrectly classified instances	52
Kappa statistic	0.7431
Mean absolute error	0.2025
Root mean squared error	0.3182
Relative absolute error	40.5498%
Root relative squared error	63.6789%

Table 6: Detailed accuracy by class in filtered classifier

Variables	TP rate	FP rate	Precision	Recall	ROC curve
Positive	0.842	0.100	0.887	0.842	0.927
Negative	0.900	0.158	0.859	0.900	0.927
Weighted average	0.872	0.130	0.873	0.872	0.927

Table 7: Confusion matrix for filtered classifier

a	b	Values
165	31	a
21	189	b

Table 8: Comparision between filtered and Naive Bayes classifier

a	b	Values
183	13	a
13	197	b

Using filtered classifier: The accuracy for training data and test data of this filter resulted with 87.89 and 56.62%, respectively. The detailed results are indicated in Table 5-7. Table 7 is the confusion matrix corresponding to this filter (Table 8).

CONCLUSION

This study elaborates how to examine sentimental analysis based on Twitter data on India's 4G network

'JIO'. The major advantage of this type of sentimental analysis is that one gets first hand opinion of a large group in one platform about a particular product, service or a matter. In this analysis, the machine learning tool, Weka has been used for classification and prediction of models utilizing filters viz., Naive Bayesian filter and filtered classifier. Naive Bayesian filter showed better accuracy than that of filtered classifier. In nut shell such sentimental analysis gives promising results which can be used for corrective actions by the manufacturers of product and providers of service. The study can be a useful tool in the most competitive and consumer friendly world for the betterment of service or product.

ACKNOWLEDGEMENT

The researcher thank Amrita school of arts and sciences for the facilities given to carry out this research.

REFERENCES

Agarwal, A., B. Xie, I. Vovsha, O. Rambow and R. Passonneau, 2011. Sentiment analysis of twitter data. Proceedings of the workshop on Languages in Social Media, June 23, 2011, Association for Computational Linguistics, Stroudsburg, Pennsylvania, USA., ISBN:978-1-932432-96-1, pp: 30-38.

Davidov, D., O. Tsur and A. Rappoport, 2010. Enhanced sentiment learning using twitter hashtags and smileys. Proceedings of the 23rd International Conference on Computational Linguistics: Posters, August 23-27, 2010, ACM, Beijing, China, pp: 241-2491.

Jandail, R.R.S., 2014. A proposed novel approach for sentiment analysis and opinion mining. Intl. J. UbiComp, 5: 1-10.

Kalaivani, P. and K.L. Shunmuganathan, 2013. Sentiment classification of movie reviews by supervised machine learning approaches. Indian J. Comput. Sci. Eng., 4: 285-292.

Nasukawa, T. and T. Nagano, 2001. Text analysis and knowledge mining system. IBM. Syst. J., 40: 967-984.

Pang, B., L. Lee and S. Vaithyanathan, 2002. Thumbs up?: Sentiment classification using machine learning techniques. Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing Vol. 10, July 6-7, 2002, Association for Computational Linguistics, Stroudsburg, Pennsylvania, pp: 79-86.

Shruti, W., S. Chandra, K.J. Liszka and C.C. Chien, 2009. Text mining for sentiment analysis of Twitter data. BCS Thesis, University of Akhin, Akhin, Moscow.

Vinodhini, G. and R.M. Chandrasekaran, 2012. Sentiment analysis and opinion mining: A survey. Int. J. Adv. Res. Comput. Sci. Software Eng., Vol. 2.