Sentiment Analysis of Tweets using Rapid Miner Tool

Sai Durga Pravallika Devisetty, Yarramneni Mownika Sai, Akash Vijay Yadav, Pellakuri Vidyullatha

Abstract*: *Social media is one of the popular source for information retrieval and also it is being used for sharing day-to-day events of our lives and the incidents which are happening around the world. In present days, social networking websites like Twitter, Facebook, etc. are used extensively for communication. So they become an important source for understanding the emotions and opinions of most of the people. In this paper, data mining techniques are used to perform sentiment analysis on the tweets that are shared by the people on Twitter. In order to achieve this, tweets are collected from Twitter, text mining techniques are applied, and then they are used for building sentiment classifier. Rapid Miner software is used for this purpose. Here, three different classifiers, namely, K-Nearest Neighbor,Naive Bayes, SVM, are applied on data and then the results obtained are compared. The SVM algorithm proved to be more accurate compared to the other two algorithms used.

Index Terms*: *Classification, RapidMiner, Sentiment analysis, Text Mining, Twitter.

I. INTRODUCTION

Data mining is the process of extracting information from very large sets of data. Huge data is not useful for us in any way unless we are not able to acquire our required information or analyze the data or predict the behavior based on given patterns[7].This should be done in lesser time and with lesser complexity and greater accuracy. Thus, data mining plays an important role in present in Information Industry. There are many applications of data mining[8], one of which is sentiment analysis. Sentiment Analysis can be performed on text documents or social media websites like Twitter, Facebook, etc. Sentiment Analysis allows users to get an idea about the opinions expressed in tweets and classifies them into positive and negative categories. The companies can use sentiment analysis to understand the customer's opinion about their products, and this will help them to improve their services in future based on the reviews to get better in their work. In 2017, MdShoeb & Jawed Ahmed, [1] implemented sentiment analyis using Decision tree, Naive Bayes, kNN algorithms and compared the accuracies and precisions of the results obtained. Shahid Shaya etal(2018) [2] used sentiment analysis and opinion mining on big data and discussed different areas of application. Xinran Zhang and Chen Li(2017)[3] built model to help government in understanding the emotions of the residents using data of economic development and social stability. Alvin Chyan etal(2012)[4] proposed a model based on the moods and stock exchanges of the previous day and a specific market order is determined to maximize investment returns. Rowan Chakoumakos etal(2011) [5] developed a system for economic analysis of stocks based on natural language processing of tweets in twitter. Ikoro etal(2018) [6] implemented sentiment analysis on opinions of energy consumers of UK by using two lexicons, for extracting sentiments and for classification of data respectively. In this paper, three algorithms, namely, Naive Bayes, SVM and kNN algorithms are applied on twitter data to find which algorithm gives better accuracy in prediction compared to the other two algorithms.

II. METHODOLOGY

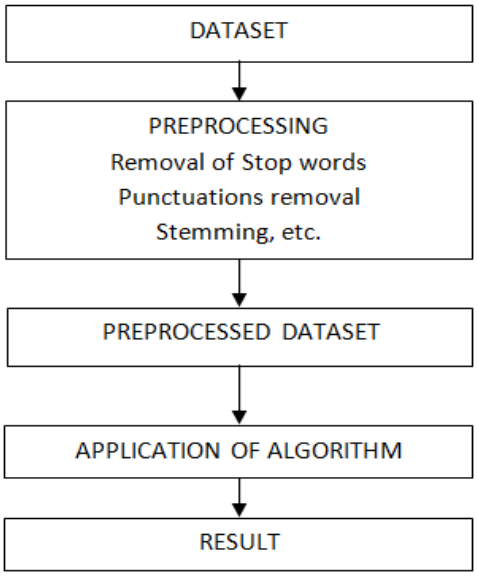

Figure 1: Algorithm

The dataset is preprocessed and algorithms are applied for getting the result.

Revised Manuscript Received on April 07, 2019.

Sai Durga Pravallika Devisetty, CSE, Koneru Lakshmaiah Education Foundation , Guntur,India.

Mownika Sai Yarramneni, CSE, Koneru Lakshmaiah Education Foundation , Guntur,India.

Akash Vijay Yadav, CSE, Koneru Lakshmaiah Education Foundation , Guntur,India.

Pellakuri Vidyullatha , CSE, Koneru Lakshmaiah Education Foundation , Guntur,India.

2.1 NAÏVE BAYES

Naïve Bayes is a classification technique based on Bayes theorem. It is used for large sets of data. It is simple as well as outperforms many other classification techniques

$$P(c/y) = \frac{P(y/c).P(c)}{P(y)}$$

Where, P(c/y) is the posterior probability of class given predictor, i.e., the probability of occurrence of c given y.
P(y/c) is the probability of predictor given y, i.e., the probability of occurrence of y given c (likelihood).
P(c) is the prior probability of class.
P(y) is the prior probability of predictor.
Here, we can easily get prior probability of class, i.e., P(c) as every class is equally probable.

Algorithm:

```
TrainNB(W,D):
    V<- getVocabulary(D)
    S<- count(D)
    for each wϵW
    do Sw<- countInClass(D,w)
        priorp[w] <- Sw/S
        for each vϵV
        do Swv<-countWithTerm(D,w,v)
            condp[v][w]<-(Swv+1)/(Sc+2)
    return V, priorp, condp
```

First the data is cleaned and tokenized. Next, the likelihood of predicting a particular word as positive or negative is calculated from the labeled set of positive and negative words[9]. Now, Naive Bayes equation is used to calculate probability and the higher probability decides if the attribute x is positive or negative. Advantages are that – it gives better results in multiclass prediction; with lesser training data, Naïve Bayes performs well compared to others like logistic regression, etc., when the assumption of independence holds. Naïve Bayes is used for doing real time predictions, can predict the probability off multiple classes, used for text classification, sentiment analysis, spam filtering and is used along with collaborative filtering in order to build recommendation systems.

2.2 SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification as well as regression purposes.
First, the data is cleaned, preprocessed and made ready for the implementation. Then, the data is plotted with the value of the point in an k-dimensional plane where k is the count of the features. After the plotting is done, a hyper plane is found in such a way that it differentiates the two classes formed, very well. The correct hyper plane should be found inorder to get accurate results. SVM is similar to the model of neural networks and radial basis functions. SVM has better regularization than that of other models.In this kind of model, over-fitting can be avoided.

```
TrainSVM(W,C):
    Order samples in ascending order
    minerr=1000; v=0
    decision yp ϵ{-1,1}
    if first wi values are of same class:
      return x
    end if
    for all w do
    train SVM
            classify with error costs
    errneg=P(y=1/yp');
    errpos=1-(y=1/yp')
    if errpos<minerr:
      minerr=errpos;v=1
    end if
    if errneg*x<minerr:
      minerr=errneg;v=-1
    end if
     end for
     return v
```

2.3 k-NEAREST NEIGHBOR

kNN algorithm is used mostly for classification and sometimes for regression purposes. For any technique to be efficient, we consider the parameters like calculation time, predictive power and ease to interpret output. kNN algorithm works fairly across all the parameters. The k in k-nearest neighbor algorithm is the number of nearest neighbors we wish to include in. For example, if we want to find the class of a particular point, we will include in the plane k number of points which are present around the particular point that we have selected. If this is possible, then we can say that the particular point belongs to the class of other k points.
The value of k that we choose plays a major role in the algorithm. With the increasing value of k, we can observe that the boundary becomes smoother. If the k value is infinity, then all the points are considered as the class which is having majority in the initial stage. When k=1, the closest point to any data is itself and thus the error rate at this value of k is always zero.

```
TrainkNN():
     Df =ϕ
     foreach dϵ D do
    DES<-kNN(d, Dkxi)
    if classification:
        F<- classify(DES)
    else if regression:
        F<-regress(DES)
        Df <-Add(s,F)
     return Df
```

First, the data is cleaned and preprocessed for application of the algorithm. Next, the data is plotted to understand the similarity between them, which can be calculated by distance on the plot. Based on the distance, we can understand if the given word is positive or negative.

III. RESULTS AND DISCUSSIONS

For this sentiment analysis purpose we have used a tool called RapidMiner, which is a data science software platform used for data preparation, machine learning, etc., purposes. A powerful solution for sentiment analysis would be to train using historical data and predict the sentiment. So, this is used in our project and accuracy of different algorithms is found out.

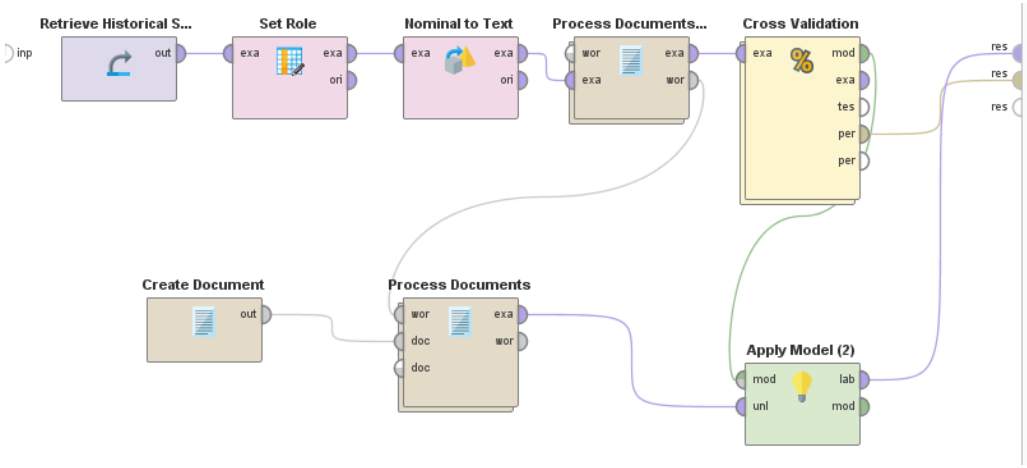

Figure 2: Block diagram

Figure 3: Validation operator with Naïve Bayes algorithm

accuracy: 62.00% +/- 6.78% (micro average: 62.00%)

	true negative	true positive	class precision
pred. negative	50	31	61.73%
pred. positive	45	74	62.18%
class recall	52.63%	70.48%	

Figure 4: Naïve Bayes accuracy

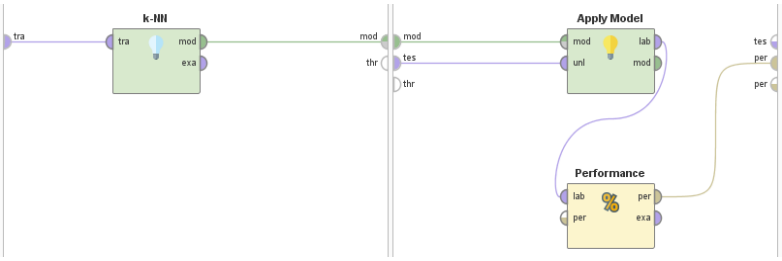

Figure 5: Validation Operator with k-NN algorithm

accuracy: 58.50% +/- 8.38% (micro average: 58.50%)

	true negative	true positive	class precision
pred. negative	57	45	55.88%
pred. positive	38	60	61.22%
class recall	60.00%	57.14%	

Figure 6: k-NN accuracy

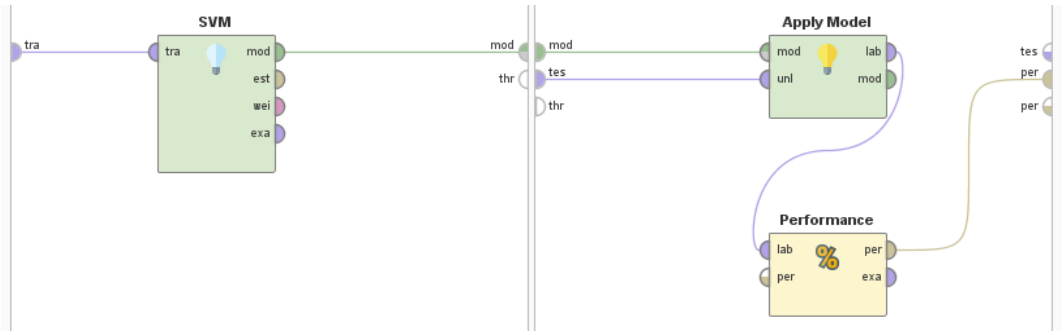

Figure 7: Validation operator with SVM algorithm

accuracy: 65.00% +/- 6.32% (micro average: 65.00%)

	true negative	true positive	class precision
pred. negative	27	2	93.10%
pred. positive	68	103	60.23%
class recall	28.42%	98.10%	

Figure 8: SVM accuracy

Figure 1 represents the main process. Here, historical data is taken and the data is processed, i.e., the data is tokenized and they are converted to a particular case. The document operator contains the tweets that we want to analyze and these tweets are also processed and model is applied.

The validation operation is used for applying different data mining algorithms and to get the result of percentage of accuracy the algorithm is giving.

We have performed the experiment with algorithms like Naïve Bayes, SVM and k-Nearest Neighbor algorithms

IV. CONCLUSION

In this, we have used three different data mining algorithms, Naïve Bayes, k-Nearest Neighbor and SVM algorithms and classified the tweets into positive and negative, with respect to the historical data. The result shows that the accuracy of Naïve Bayes is 62.00%, the accuracy of k-NN algorithm is 58.50% and the accuracy of SVM algorithm is 65.00%. So, from the results, we can understand that, compared to Naïve Bayes and k-NN algorithms,

SVM algorithm can be used to get better results as the accuracy obtained in the case of usage of SVM algorithm is higher compared to Naïve Bayes and k-NN algorithms.

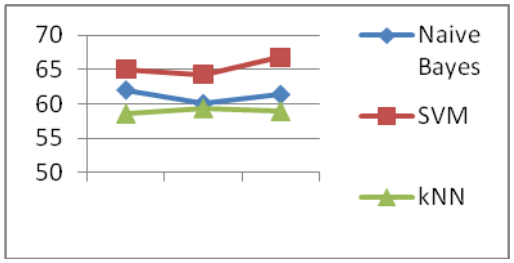

Figure 9: Comparision of algorithms

REFERENCES

1. MdShoeb, Jawed Ahmed, "Sentiment Analysis and Classification of Tweets Using Data Mining", IRJET, Volume:4,Issue:12,e-ISSN:2395-0056, 2017.
2. Shahid Shaya, Noor ismawati jaafar, shamshul Bahri, AInin Sulaiman, Phoong Seuk Wai, Yeong Wai Chung, Arsalan Zahid Piprani,Mohammed Ali Al-Garadi, "Sentiment Analysis of Big data: Methods, Applications, and Open Challenges", IEEE Access,ISSN:2169-3536, 2018 ,Pg:37803-37827.
3. Xinran Jhang, Chen Li, "The research of sentiment analysis of microblog based on data mining exampled by basic endowment insurance", IEEE,2017.
4. Alvin Chyan, Tim Hsieh, Chris Lengerich, "A Stock-Purchasing Agent from Sentiment Analysis of Twitter".
5. Rowan Chakoumakos, Stephen Trusheim, VikasYendluri, "Automated Market Sentiment Analysis of Twitter for Options Trading."
6. Victoria Ikoro, Maria Sharmina, Khaleel Malik, Riza Batista-Navarro,"Analyzing sentiments expressed on twitter by UK Energy Company Consumers", Fifth International Conference, IEEE,2018.
7. D.Rajeswara Rao, Vidyullatha Pellakuri, "Knowledge Based Information Mining on data using statistical approaches", International Journal of Pharmacy and technology, Volume 8, issue 4, December 2016, pg no:21961-21966.
8. B.Sekhar Babu, P.Lakshmi Prasanna, P.Vidyullatha, "Personalized web search on e-commerce using ontology based association mining", International journal of Engineering and technology, Volume 7,issue 1.1, 2018,Pg no:286-289.
9. Moulana Mohammed, "Incremental Learning Decision Tree Algorithm for Knowledge Discovery", PONTE Journal, Volume 72, Issue 8,2016, Pages:163-170.