Real Time Big Data Sentiment Analysis and Classification of Facebook

Watheq Ghanim Mutasher
Information Institute for Postgraduate Studies, Iraq.
E-mail: ms201920531@iips.icci.edu.iq

Abbas Fadhil Aljuboori
College of Engineering, University of Information Technology and Communications, Iraq.
E-mail: abbas.aljuboori@uoitc.edu.iq

Received August 22, 2021; Accepted December 02, 2021
ISSN: 1735-188X
DOI: 10.14704/WEB/V19I1/WEB19076

Abstract

Many peoples use Facebook to connect and share their views on various issues, with the majority of user-generated content consisting of textual information.

Since there is so much actual data from people who are posting messages on their situation in real time thoughts on a range of subjects in everyday life, the collection and analysis of these data, which may well be helpful for political decision or public opinion monitoring, is a worthwhile research project.

Therefore, in this paper doing to analyze for public text post on Facebook stream in real time through environment Hadoop ecosystem by using apache spark with NLTK python.

The post or feeds are gathered form the Facebook API in real time the data stored database used Apache spark to quick query processing the text partitions in each data nodes (machine). Also used Amazon cloud based Hadoop cluster ecosystem into processing of huge data and eliminate on-site hardware, IT support, and other operational difficulties and installation configuration Hadoop such as Hadoop distribution file system and Apache spark.

By using the principle of decision dictionary, emotion analysis is used as positive, negative, or neutral and execution two algorithms in machine learning (naive bias & support vector machine) to build model predict the outcome demonstrates a high level of precision in sentiment analysis.

Keywords

Apache Spark, Machine Learning, Big Data, Real-Time, Sentiment Analysis, Facebook.

Introduction

Social networking has been rapidly changing over the last few years, and it now affects every area of life. People continue to post content on Facebook, which is one of the most popular social media networks (Kaur *et al.*, 2019). They often provide input on the post of their choice by liking, commenting, or debating it. As a result of these likes, comments, and updates, a massive amount of data is gathered, resulting in Big Data (Kaur *et al.*, 2019).

Sentiment Analysis is an important part of Big Data, which is a common research field of Computer Science. Big Data is described as a huge amount of data that is easily accessible via the internet, social media, and other means (Science *et al.*, 2015).

Sentiment analysis is the process used of founding the text's polarity, or the user's perspective on a specific subject. Sentiment analysis is extremely useful in determining a person's attitude toward a specific subject (Rodrigues and Chiplunkar, 2018).

In addition, all internet data would be in a format that is either unstructured or semi-structured. The greatest problem, it turns out, is effectively processing data of this nature so that information can be extracted and collected for further surveying (Rodrigues and Chiplunkar, 2018). Traditional data analysis methods have increasingly lost their ability to analyze massive data sets effectively. Hadoop tools has recently indicated to be a strong platform for processing massive data sets (Rodrigues and Chiplunkar, 2018). Apache Spark is a distributed computing platform that is open source. It's built to handle reduced tasks while also storing data and outcomes in memory. As a result, it's known as Memory Computing, and it boosts data processing speed. As a result, Spark outperforms Hadoop when it comes to data mining and machine learning (Albaldawi and Almuttairi, 2020). The RDD (Resilient Distributed Dataset) function of Apache Spark is superior, as it provides fault-tolerant artifacts that may be used deployed over a parallel computer (Albaldawi and Almuttairi, 2020). An RDD is a read-only, partitioned array of records that is immutable. Only deterministic operations on either (1) data in stable storage or (2) other RDDs can be used to build RDDs. The Spark RDD will run on top of Hadoop, taking advantage of Hadoop Map Reduce Worlds' Resilient in Memory computations (Dasgupta *et al.*, 2015). The Sentiment Package was used to conduct sentiment analysis in this case. The Sentiment Package uses a subjectivity dictionary to calculate sentiment. For situations where the text is from a social media platform and is in English, the dictionary is reasonably robust. The polarity (positive or negative subjectivity) of the dictionary words is often correlated with it (Dasgupta *et al.*, 2015).

Textual posts' sentiment can be broken down into three categories.
1-Level of Document:
Analytical analysis of the entire document to evaluate its polarity (positivity, negativity, or neutrality) on the topic.
2-Level of sentence:
Each sentence is subjected to an analysis to determine whether it is positive, negative, or neutral.
3- Level of Aspect:
Analytical research is out at a fine granular level. Each aspect or feature of the product is examined, and polarity for each feature is calculated (Sharma, 2018).

The techniques for performing sentiment analysis are as follows.

1- A Methodology for Machine Learning:

Sentiment analysis can be performed using a variety of machine learning methods, such as Nave Bayes. The creation of classifiers from tagged instances of textual posts is the focus of these learning methods. They are successful in the field in which they were educated (Sharma, 2018).

2- A Lexicon-Based Methodology:

These techniques use the sentiment score of the document's words or phrases to assess the document's emotional orientation. These lexical techniques make use of a dictionary with a large number of terms annotated with their polarity, strength, and semantic orientation (Sharma, 2018).

Related Works

(Poecze, Ebster and Strauss, 2018), Proposed a Social media metrics and sentiment analysis to measure the impact of social media messages using Machine learning to enhance brand communication and analyze consumer reaction on the YouTube Gamers Facebook page.

(Ramanathan and Meyyappan, 2019), proposed Using Lexicon-based Feedback on Oman Tourism on Twitter, offered a Twitter Text Mining for Sentiment Analysis on People's Feedback on Oman Tourism.

(Kaur *et al.*, 2019), The aim of the study is to conduct a comparison of news data obtained from the Facebook graph API and analyzed using MongoDB and Hive. The data is contained in a NoSQL database, MongoDB, and the Hadoop ecosystem's Hive data

warehouse. MongoDB has a lower execution time for identical queries than Hive due to limited resources and vast amounts of unstructured data.

(Science *et al.,* 2015), The suggested approach uses a dictionary to analyze emotions and categorize text as positive or negative.
Each tweet's emotions are determined by assigning it a degree of polarity. It is considered a positive feeling if it is greater than zero, but it is considered a negative feeling if it is less than zero. Extract sentiment using codes and hashtags using a modern, business-friendly approach to big data that is powerful and scalable.

(Dasgupta *et al.*, 2015), Analyzing brand sentiment from Facebook data using open source technology. The work that was done to evaluate consumer sentiments on various brands using Facebook data. The framework offers an analysis of information top level of Hadoop, supporting the Map Reduce paradigm and the scalability of Spark RDDs, as well as an advanced statistical analysis layer via R, with open source stack architecture and run time optimizations.

(Kılınç, 2019), The systems' sentiment classification performances for offline and real-time modes, respectively, are 86.77 percent and 80.93 percent, using the author nave bias model from the apache spark machine learning package. He demonstrated a real-time spark-based SA framework with the following software components: I spark machine learning and streaming service; (ii) Twitter streaming service; (iii) Twitter fake account detections service; and (iv) real-time monitoring and panel software.

Hadoop Framework

Apache Hadoop is a free and open source program platform that employs programming methods to enable the processing of massive data sets across multiple clusters of commodity servers. It's built to scale from a single computer to a large number of computers, all while maintaining a high level of fault tolerance.

Hadoop architecture includes Map-reduce programming model, Hadoop distributed file system (HDFS), and ecosystem components such as Hive, Pig, Hbase, Scoop, and others for handling big data. Hadoop offers a number of methods for processing big data, which are referred to as ecosystem components as illustrate in Fig.1(Jadhav, Patankar and Jadhav, 2018).

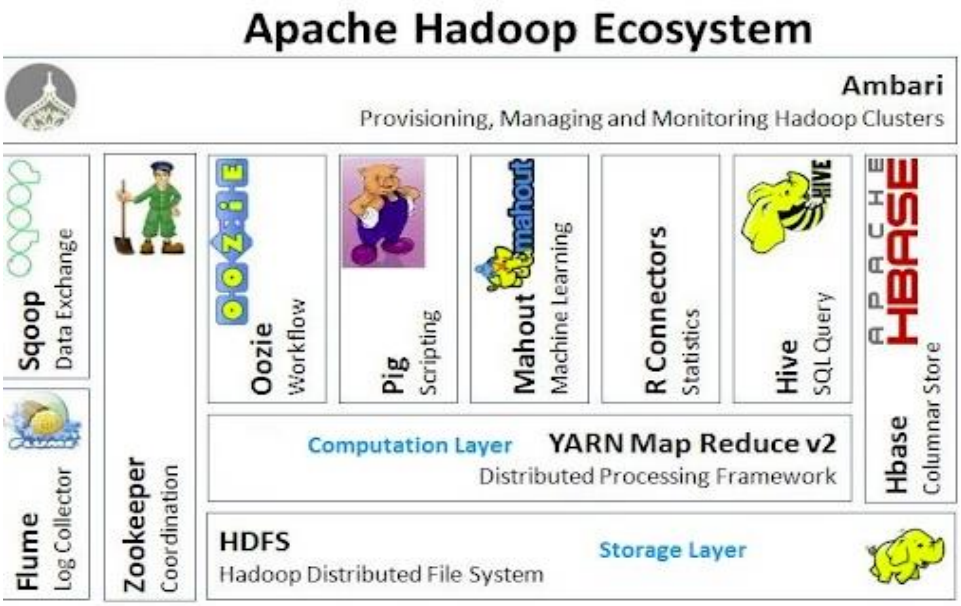

Figure 1 Hadoop Ecosystem components (Jadhav, Patankar and Jadhav, 2018)

1. HDFS (Distributed File System)

HDFS is a Hadoop file system execution of a distributed file system architecture that can carry a large number of data and make available a simple way for several clients around the network to access it. It's made to run on cheap hardware and is extremely fault-tolerant (called commodity hardware). HDFS is a file system with a block structure in which files are broken down into fixed-size blocks and loaded on Hadoop clusters. The HDFS uses Name and Data nodes to implement a Master-slave architecture. Multiple Data nodes served as slaves while the Name node acted as a master (Jadhav, Patankar and Jadhav, 2018).

- **Map Reduce (Distributed Data Processing)**

The Map-Reduce system model is used to process large amounts of data. The Map/Reduce algorithm employs the divide-and-conquer approach, in which a large complicated issue is broken down into smaller sub-problems that are solved simultaneously and independently through distributed clusters. Finally, all of the sub-solutions were combined to produce the final solution to the problem (Jadhav, Patankar and Jadhav, 2018).

- **Apache Spark**

Apache Spark is a distributed computing engine that can run on a variety of workloads and platforms. Spark uses a range of paradigms to bind to various networks and process different data workloads, including Spark Streaming, Spark ML, Spark SQL, and Spark

Graph x. Apache Spark is a quick in-memory information processing engine with elegant and descriptive development APIs that allow data workers to execute streaming machine learning or SQL workloads that require fast interactive access to data sets efficiently (Ramanathan and Meyyappan, 2019). Real-time streaming, queries, machine learning, and graph processing are all available in Spark. Various techniques were used for different types of workloads before discovered Apache Spark. Batch analytics, interactive queries, real-time streaming processing, and machine learning algorithms all have their own databases. Instead of relying on different technologies that aren't always compatible, Apache Spark can handle all of these tasks (Alla, 2018).

Sentiment Analysis

Sentiment is characterized as an author's expression or opinion on any object or part. The main focus of sentiment analysis is parsing the text and finding the opinion word. After identifying opinion words, sentiment values are assigned to these words. Finally, must determine the text's polarity. Positive, negative, or neutral polarity exists. For sentiment classification, used the Lexicon method. The sentence is split into words as part of the parsing process. Tokenization is another name for this move. These tokenized words are used to classify opinion words as an input (Science *et al.*, 2015).

Methodology

The proposed system is used to work on data in real time which fetch by Graph API, this particle real time failed with traditional analysis tools, therefore used here tools very strong to deal with huge data stream form Facebook on cloud AWS and used NLTK which part form NLP to analysis text into positive and negative. Also apply more than one algorithm to measure accuracy of our model.

1. Registration by Graph API

Used the Facebook API is a set of tools that are used to bring data into and out of the network. On the developer's site, you must register or sign up with your existing Facebook account, but your identity must be verified. It'll just take a few easy steps to accomplish this:

- **Assign Node, Edge, and Connection**

Nodes are generally often the starting point for reading operations. A node is a single item with its own identifier. There are numerous User node objects on Facebook, for example, each with a unique ID that represents a person. The ID of a certain item is used to read a node.

Nodes have edges, which can often return groups of other nodes related to them. You must provide both the node ID and the edge name in the path to read an edge. The /feed edge on /user nodes, for example, can retrieve all Post nodes on a User. During the Receive access token flow, you'll need to get a new access token and select user posts rights.

Connection are the attributes of nodes. By default, when you query a node, it returns a collection of connection. However, by using the connections argument and naming each connection, you can choose which connections you want returned. This changes the defaults and only returns the fields you specify plus the object's ID, which is always returned.

Nodes are typically used to obtain data about a single item, edges are used to obtain collections of objects on a single object, and fields are used to obtain data about a single item or each object in a collection.

- **Sentiment Analysis Process**

Wrote a python script which:

1. Facebook integration.
2. Collect posts and comments from Facebook using Graph API.
3. Write the Facebook posts and comments to a CSV files after removes duplicates, cleans invalid values in real time, and selects relevant comments.
4. Uses the NLTK (Natural Language Toolkit) Vader libraries to perform sentiment analysis.
5. Build model by two algorithms (Naive bias, support vector machine) to training and testing data to get high accuracy to our model with draw predication compassion with line data.
6. Generates output chart for, showing sentiment analysis scores. Each chart is saved as an image file.
7. generates a csv file containing sentiment analysis scores across.
8. Matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms as illustrate in Fig (2).

- **Stream Dataset from Facebook**

It was read post and work composition for (id page) with (id post) to fill array by post and duplicate every time. After full all array by post must be define two array one for comment others into time comment. Become found us parameter for comment assign to id post which will give us comment answer on this post.

In each time get on 1000 comment, although array not fill will continuous read post in store in HDFS Hadoop as illustrate in Fig (2):

After this must be split into text comment and time comment which processing text comment as shown below by NLTK with apache spark and the same time draw time comment.
NLTK with distributed processing by apache spark used to classify all comments into positive and negative as illustrate in Fig (2).

Now will sent comment array to analysis and time array to graph after this used NLTK libraries in python to read post before storage in real-time**.**

1. Preprocessing: Convert characters into lowercase and determine idioms, this step called preprocessing where Facebook contain more field such as id, text, language, time…etc. will extract text into sentiment analysis and used to remove all stop word and punctuation, noise, capital letter…. etc.
2. Segmentation into sentence.
3. Tokenization I: split each sentence into multiple word based on space.
4. Emoticon detection: recognizing the pictorial representation of a facial expression.
5. Tokenization II: complete.
6. Interjection detection: part of speech that demonstrates the emotion or feeling (like Ahh, that feels wonderful).
7. Token score assignation: specify scoring depend on new words.
8. Syntactical analysis: determining the logical meaning of sentence or part of sentence.
9. Polarity calculation.

Noise from post had been removed stored in HDFS. Then data processed in apache spark which contains four nodes one as Master node and others as data node by distributes processing in same time to all node.

Must have to split the sentence to words this process pf splitting up of stream of text into words is said to be tokenization and this regard step fed as input for furthers processing of sentiment analysis.

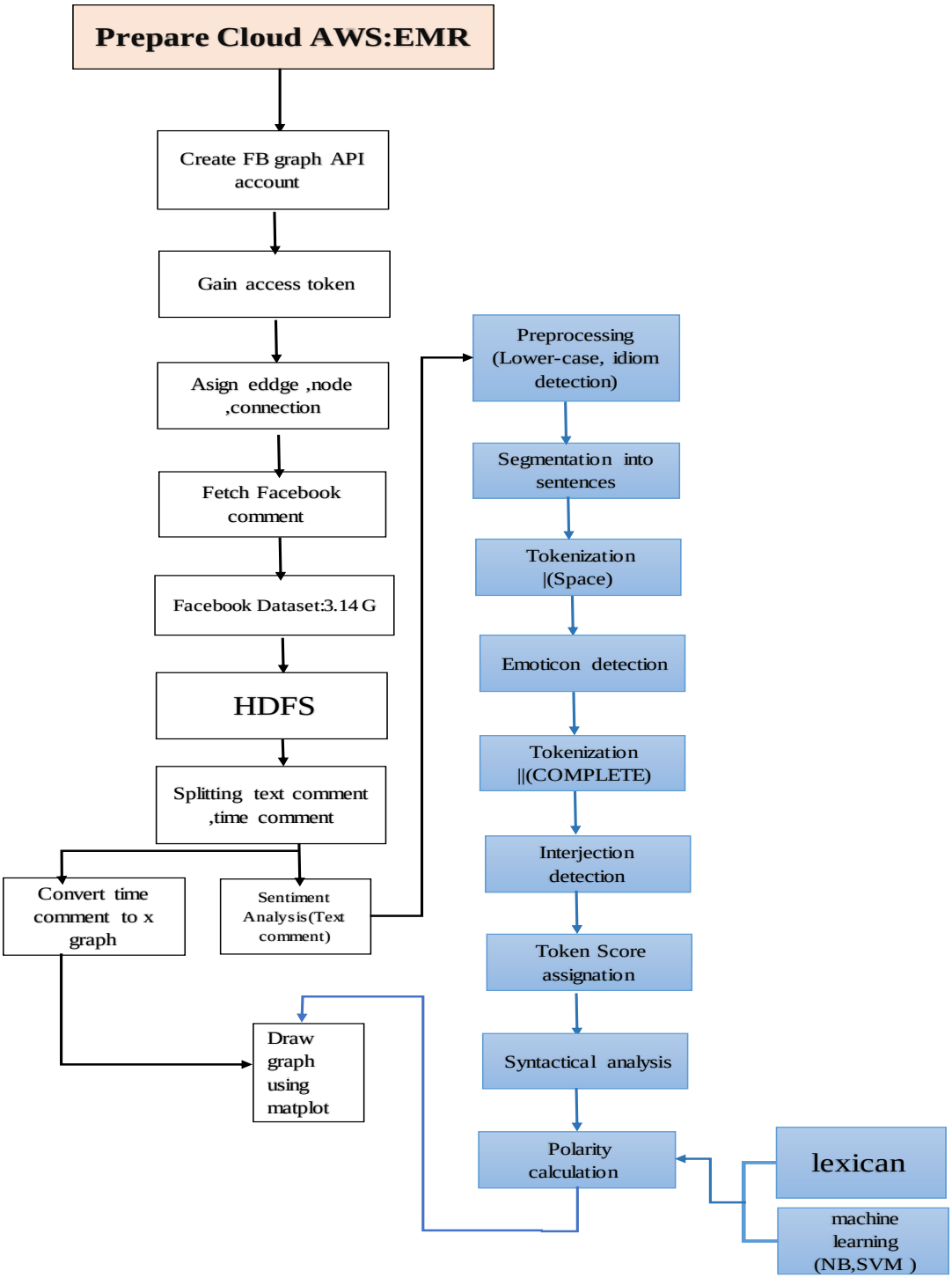

Figure 2 Sentiment Analysis Diagram

- **Hybrid Solution**

Lexicon in NLTK with Machine learning, here must be accuracy of model high accuracy. Must take post and using NLTK to give polarity for each word to classify it into positive

and negative. NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.

Finally, that need to detect the polarity of the text, polarity can be positive, negative. And need lexicon in NLTK form sentiment classification. As parsing step, the sentence is into word the step called tokenization. These tokenization word are taken as an input for identifying opinion word. then performed sentiment analysis of real time post by using apache spark.

To perform Scoring upon NLTK as illustrate in flowchart below in [Fig. 3].

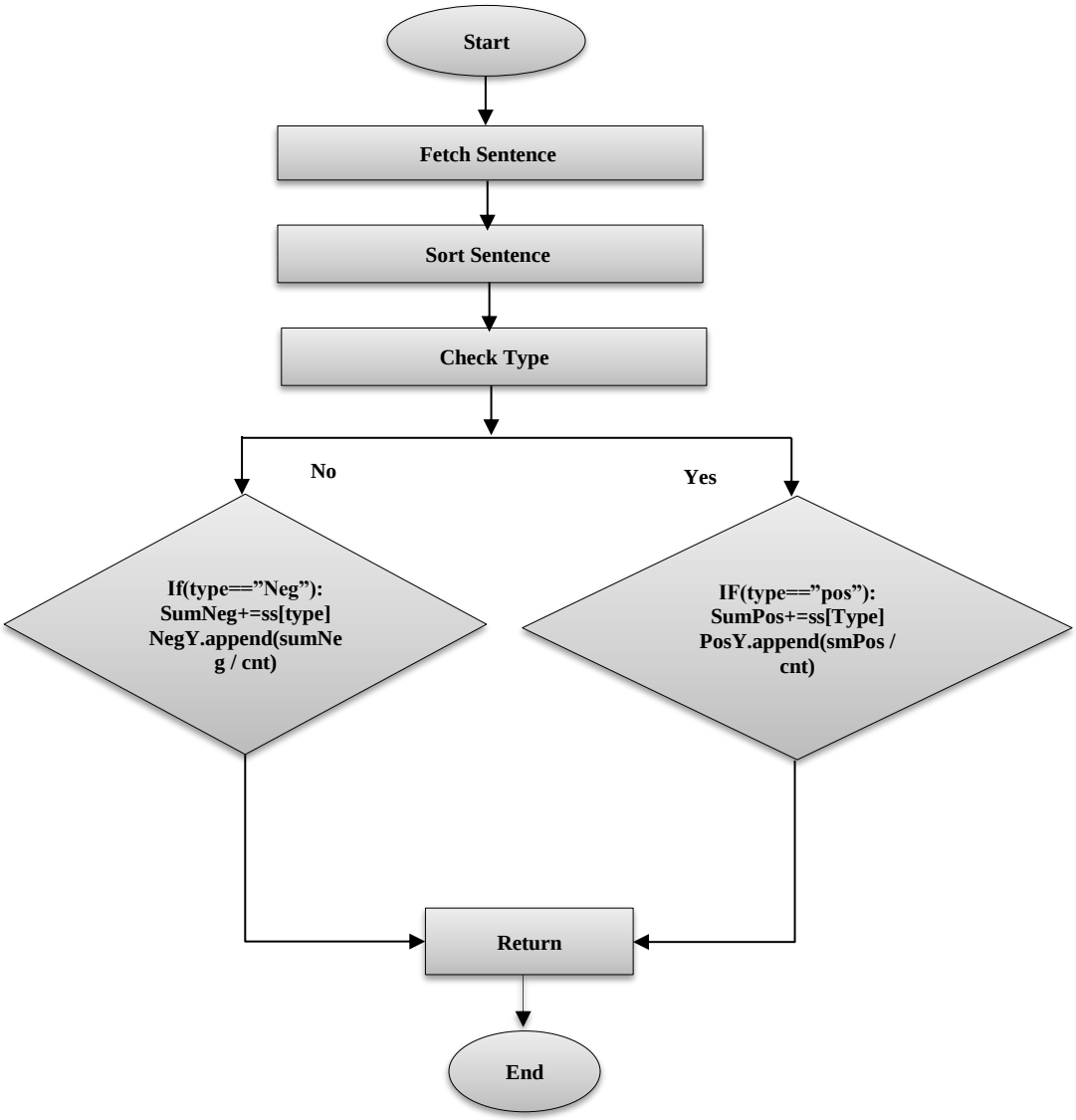

Figure 3 Flowchart Scoring Positive & Negative

Result

The live score by using NLTK Polarity score for sentiment analysis (positive, Negative), where x axis present the time and y axis represented score of comment as illustrated in [Fig. 4].

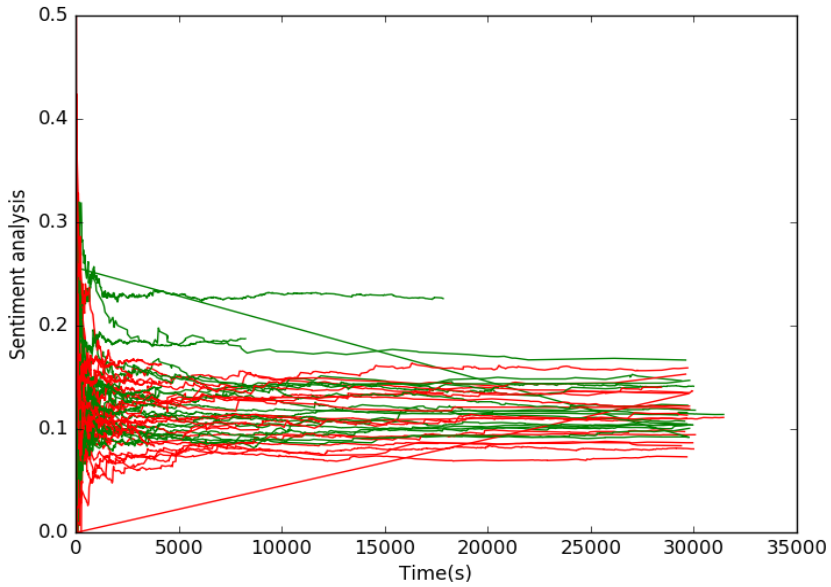

Figure 4 Sentiment Analysis for Comment positive and Negative

Green line represents Positive.

Red line represents Negative.

The live score by using NLTK polarity score for sentiment analysis only positive as illustrate in [Fig. 5].

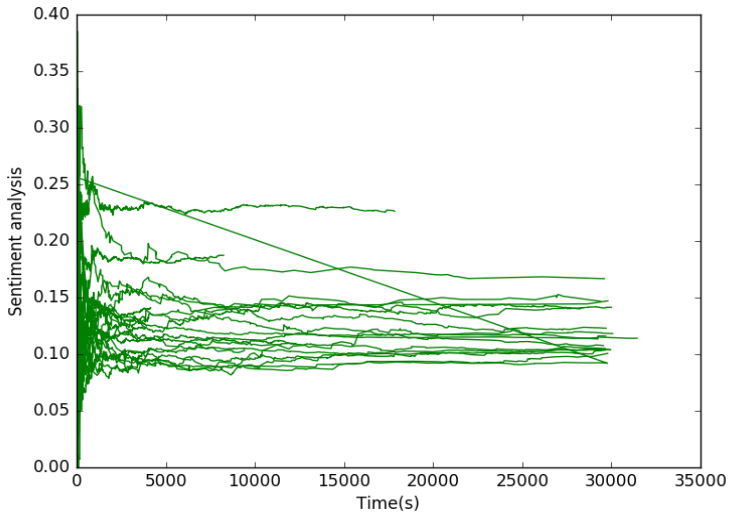

Figure 5 Sentiment Analysis for Comment Only Positive

The live score by using NLTK polarity score for sentiment analysis only positive as illustrate in [Fig. 6].

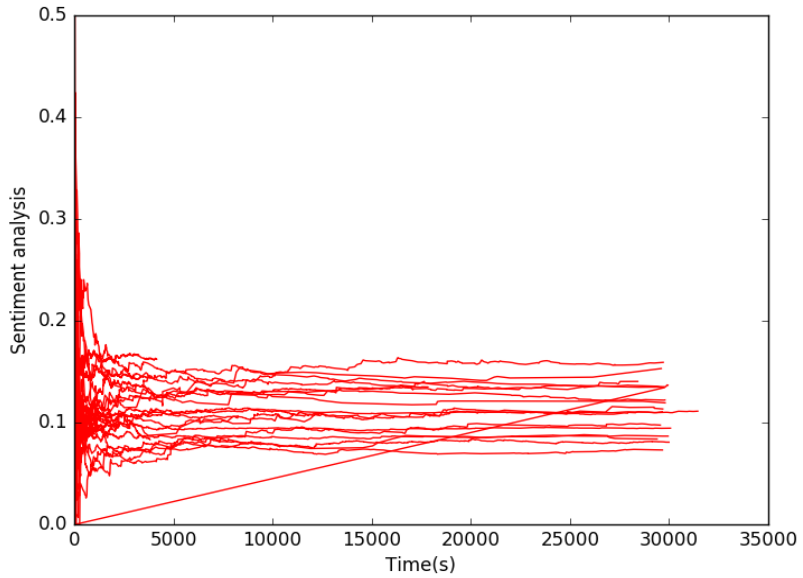

Figure 6 Sentiment Analysis for Comment Only Negative

- **Apply Two Algorithm Naïve Bias and Support Vector Machine**

It's now time to pick an algorithm, divide our data into training 80% and testing sets 20 %, and get started! The Naive Bayes classifier will be the first algorithm will use. This is a very common text classification algorithm.

The first 80% of shuffled reviews as the training package, which will include both positive and negative reviews. Then compared the results and the final 20% to see how accurate the model.
Started by simply invoking the Naive Bayes classifier, and then use Train () to train it all in one line.

It's simple enough now that it's been educated. Then can put it to the test:

Since still have the correct answers, can "test" the data. As a result, while checking, display the data to the machine without providing the correct response. If it correctly guesses what knew the answer to be, the machine is right. Given the shuffling have done, can arrived at different accuracy.

It was obtained from the model accurate equal (**87.24)** and the real comment sentiment with the prediction comment sentiment as illustrate in Fig (10).

Describe the ratio between real time of sentiment analysis and the prediction comment sentiment analysis the accuracy using naïve bias as illustrate in [Fig. 7].

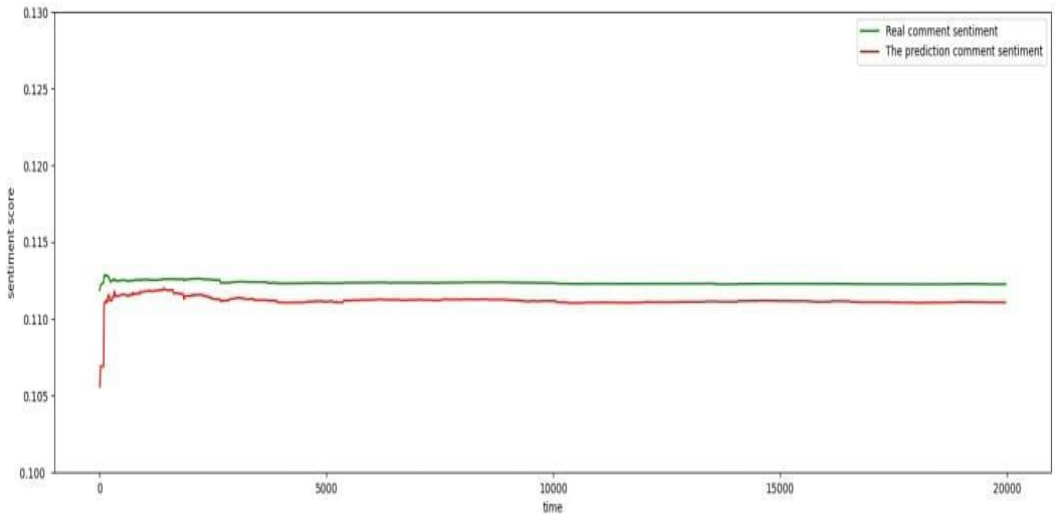

Figure 7 Real Time Sentiment Analysis and Prediction Comment Sentiment

The accuracy of our model which build by algorithm naïve bias as illustrate in table below.

Table 1 Accuracy of Model Build by Naïve Bias Algorithm

algorithm	Accuracy
Naïve bias	87.24

After Classification Positive and Negative by NLTK

Apply another algorithm support vector machine and dividing data classify into 80% as training and 20% as testing.

Used the first 80% of shuffled reviews as the training package, which will include both positive and negative reviews. Then can compare the results and the final 20% to see how accurate the model.

The accuracy that get is ~ (**0.83632369095569392**)

The figure below illustrates draw line predication in real time by using SVM on apache spark after classification dataset to positive and negative by NLTK as illustrate in [Fig. 8].

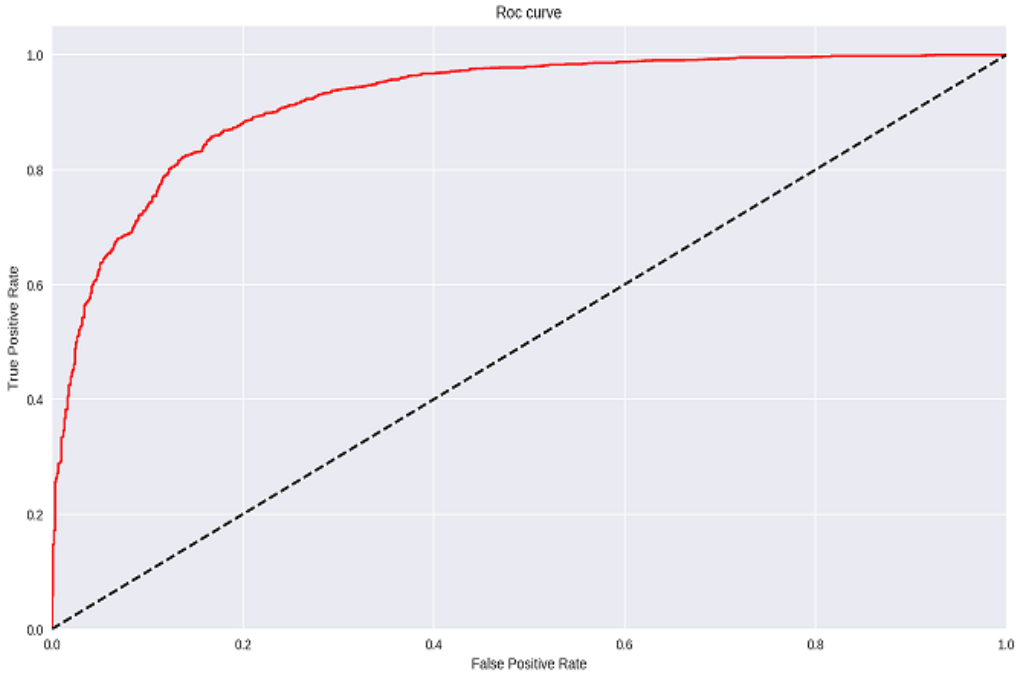

Figure 8 Show line prediction in real time by SVM

The accuracy of our model which build by algorithm support vector machine as illustrate in table below.

Table 2 Accuracy of Model Build by SVM Algorithm

Algorithm	Accuracy	F1	Precision	Recall
Support vector machine	0.836323690955693 92	0.871144422023637 84	0.915202907328891 54	0.831133113311331 11

As a result, suggest that the proposed technique produce fine performance for many types of sentiment analysis. Even if our technique depends upon Facebook to detect each particular type of comments (positive, negative), the results of experiments on real computer show the proposed technique is capable of detecting positive and negative in real time. Our proposed technique does not require a high computing resources, and it can detect user feedback that occur during each time interval. Our results also suggest that in many cases, the performance of feature combination was inferior than the performance of single feature. In addition, the comparison between using raw features and modified features strongly suggests that feature selection has an major effect on performance of our technique. In some cases, Performance is improved by using modified features instead of using raw features.

Conclusion

The ultimate goal of our research is to develop a highly flexible technique to automatically detect user feedback related to specific topic (positive or negative) in real time. Has been suggested a feasible technique that has various flexible capabilities for performing this difficult task. Experiments have been done with real network traffic and compared the detection performance of three well-known lexicon based analysis using NLTK. Also tried to improve the quality of the features by using NLTK dictionary to remove noise from the raw features. The results show that our proposed technique performs well in task of sentiment analysis and has a good possibility for applying in real-time system for different type of data (business, learning, etc.). Never the less, indicating time consumption of our technique during the training and test phase are need. Another challenge for our future research is to provide a cost effective cloud platform for academic purposes.

References

Albaldawi, W.S., & Almuttairi, R.M. (2020). Comparative Study of Classification Algorithms to Analyze and Predict a Twitter Sentiment in Apache Spark. *IOP Conference Series: Materials Science and Engineering*, *928*(3). https://doi.org/10.1088/1757-899X/928/3/032045

Alla, S. (2018). *Hadoop 3*. www.packtpub.com

Dasgupta, S.S., Natarajan, S., Kaipa, K.K., Bhattacherjee, S.K., & Viswanathan, A. (2015). Sentiment analysis of Facebook data using Hadoop based open source technologies. *In IEEE international conference on data science and advanced analytics (DSAA)*, 1-3. https://doi.org/10.1109/DSAA.2015.7344883

Jadhav, B., Patankar, A.B., & Jadhav, S.B. (2018). A Practical approach for integrating Big data Analytics into E-governance using hadoop. *Proceedings of the International Conference on Inventive Communication and Computational Technologies, ICICCT 2018*, *(ICICCT)*, 1952–1958. https://doi.org/10.1109/ICICCT.2018.8473353

Kaur, P., Dabas, C., Singhal, V., Nangru, S., & Sehgal, A. (2019). News Data Analysis from Facebook Through MongoDB and Hive. *In Fifth International Conference on Image Information Processing (ICIIP)*, 454-458. https://doi.org/10.1109/ICIIP47207.2019.8985873

Kılınç, D. (2019). A spark-based big data analysis framework for real-time sentiment prediction on streaming data. *Software - Practice and Experience*, *49*(9), 1352–1364. https://doi.org/10.1002/spe.2724

Poecze, F., Ebster, C., & Strauss, C. (2018). Social media metrics and sentiment analysis to evaluate the effectiveness of social media posts. *Procedia Computer Science*, *130*, 660–666. https://doi.org/10.1016/j.procs.2018.04.117

Ramanathan, V., & Meyyappan, T. (2019). Twitter text mining for sentiment analysis on people's feedback about Oman tourism. *4th MEC International Conference on Big Data and Smart City, ICBDSC 2019*, 1–5. https://doi.org/10.1109/ICBDSC.2019.8645596

Rodrigues, A.P., & Chiplunkar, N.N. (2018) 'Real-time Twitter data analysis using Hadoop ecosystem. *Cogent Engineering*, *5*(1). https://doi.org/10.1080/23311916.2018.1534519.

Kurian, D.D.M.K., Vishnupriya, S., Ramesh, R., Divya, G., Divya, D., Kurian, M.K., & Divya, D. (2015). Big data sentiment analysis using hadoop. *International Journal for Innovative Research in Science and Technology*, *1*(11), 92-96.

Sharma, D. (2018). Study of sentiment analysis using hadoop. In *Big Data Analytics*, 363-376, https://doi.org/10.1007/978-981-10-6620-7_35