Location Wise Opinion Mining of Real-Time Twitter Data using Hadoop

Farha Naaz
Dept. of CSE, NIIST,
Bhopal, M.P.

Rajesh Boghey
Dept. of CSE, TITE,
Bhopal, M.P.

Sandeep Rai
Dept. of CSE, TITE,
Bhopal, M.P.

ABSTRACT
Opinion Mining is the process of detecting the contextual polarity of text. In other words, it reflects a piece of writing that is positive, negative, or neutral. The opinions of others seem to be crucial in decision making. Compressing out the usable content from these opinion sources becomes a perplexing task. Today social networking data is the best and accurate source for gathering public opinions. A large volume of data is generated everyday online which is not easy to handle and processed by traditional methods. In this research, a methodology is discussed which allows interpretation of real-time Twitter data in opinion mining. We take Twitter data because on Twitter huge opinions are shared. The analysis was done on tweets about iphone8. For this, we can fetch real-time Twitter data by using flume and storingit in HDFS. Hadoop is a best open-source solution for storing and processing a large amount of data. Hadoop has two separate components HDFS for storage and MapReduce for processing. We can integrate Apache Pig with Flume for analyzing the sentiment on the basis of location because opinions are changing from location to location. Apache Pig is used for analysis as it is best suited for both structured and unstructured data.

Keywords
Sentiment analysis, Hadoop, Apache Flume, Pig, Location-based, Big data.

1. INTRODUCTION
Sentiment analysis is also known as opinion mining. Opinion mining is helpful for companies to get business insights. The process of recognizing and categorizing opinions expressed in a piece of text computationally is known as opinion mining. It is beneficial in determining the user's attitude towards a particular topic or a product. Sentiment Analysis or Opinion Mining is the process of detection of the discourse polarity of text. In other words, it reflects that a piece of writing is positive, negative, or neutral.

Sentiment analysis is extremely useful in social media surveillance as it allows us to gain an overview of the broader public opinion behind certain topics. In this research, we have scrutinized a large data set from which we tried to determine the popularity of a given product in several locations. In order to achieve this, we examine tweets from Twitter. Tweets are the eventual source of information mainly because people tweet about everything they do including buying new products and reviewing them. In the research, Hadoop and its component i.e. Flume and Pig are used to store and process large and unstructured data sets.

1.1 Hadoop
Hadoop is an open-source, distributed computing framework developed and maintained by the Apache Software Foundation wrote in java. One of the most significant attribute of Hadoop is that it fission the computation and data across multiple nodes and then form the application computation run in parallel on these nodes. It has two separate component for storage (HDFS) and processing (MapReduce).

1.1.1 HDFS
HDFS is a file system thar builds on the existing file system. It is a Java-based sub-project of Apache Hadoop. HDFS provides scalable and reliable data storage on commodity hardware. A master/slave architecture is used by HDFS. In the architecture, HDFS has a single NameNode and more than one DataNodes. The NameNode manages the file system and stores the metadata. It acts like a file manager on HDFS. Because all files and directories are represented on the NameNode. DataNodes stores the part of the data. A file is split into one or more blocks (default 64MB or 128MB) and that blocks are stored in DataNodes.

1.1.2 MapReduce
MapReduce is a computer programming model used for processing and creating large data sets with a parallel, distributed algorithm on a cluster. A MapReduce job is specifically divide the input data set into an independent blocks that are processed by the map tasks in a completely parallel manner. The First step is the mapping of the data set in MapReduce architecture. The framework sorts the outputs of the mapping process, which are then inputted to the second step is to reduce the task. Input and the output of the job are stored in a file-system. The MapReduce framework consists of two processes which are JobTracker and TaskTracker. The JobTracker manages the resources that are TaskTracker. The TaskTracker is a processing node in the cluster. It accepts several tasks like map-reduce and shuffles from a Job Tracker.

1.1.3 Flume
Flume [4] may be a framework that is employed to maneuver log knowledge into HDFS. Usually, events and log knowledge area units generated by the log servers, and these servers have Flume agents running on them. These agents receive the information from the information generators. The data in these agents are collected by the associate intermediate node called Collector. Similar to agents, there are often multiple collectors in Flume. Finally, the information from these collectors is aggregates and pushed to a centralized store HDFS.

1.1.4 Pig

Apache Pig is a big data analytical tool through which we can analyze the large data sets over the Hadoop framework. Apache pig support structure as well as unstructured data both and also work with the Hadoop and without Hadoop. Pig queries are written as similar to SQL queries.

2. LITERATURE REVIEW

Ankur Goel [1], In his research take twitter data for analyzing users opinion about any product or service. The paper contains the implementation of Naive Bayes using sentiment140 training data using theTwitter database. Tweets can be classified into different classes. For actual implementation of this system python with NLTK and python-twitter APIs are used.

Mrunal Sogodekar [2], discussed Big Data Analytics: Hadoop and Tools. The paper focuses on a comparison of packages used in an analysis like R, Matlab, Excel, Sas, Stata.

Aditya Bhardwaj [3], In his research a comprehensive study of major Big Data emerging technologies by highlighting their important features and how they work, with a comparative study between them is presented. The paper also represents the performance analysis of Apache Hive query for the execution of Twitter tweets in order to calculate Map Reduce CPU time spent and total time is taken to complete the job.

Can Uzunkayaa [5], In his research use Hadoop and its ecosystem and implementation of the Hadoop-based platform for analyzing collected tweets. Hadoop enables exploring and processing massive and complex data. It is an open-source framework written in Java that supports parallel and distributed data processing and is used for reliable storage of data[2][5][9]. The main advantage of using it since it is cost-effective.

M. Trupthi [9] in his paper provides an interactive automatic system that predicts the sentiment of the review/tweets of the people posted in social media using Hadoop, which can process a huge amount of data. The proposed system extracts the data from SNS services which is done using Streaming API of Twitter. The extracted tweets are loaded into Hadoop and it is been pre-processed using the map reduce. This task is followed by classification which uses NLP and machine learning techniques. The classification used here is uni-word naive Bayes classification.

The research by Anwar Hridoy [14], discussed a methodology that allows utilization and interpretation of Twitter data to determine public opinions. The Analysis was done on tweets about the iPhone 6. Feature specific popularities and male-female specific analysis have been included. Mixed opinions were found but general consistency with outside reviews and comments was observed. Mining public opinion is very beneficial for the growth of the business. Opinion mining helps in market analysis before launching any product. But mining opinion along with location gives a more generalized idea about the topic.

3. PROBLEM DEFINITON

The analysis of Twitter data provides a real scan or fully completely different user opinions. Regarding what they assume and to analyze these data offer a stronger approach to make any decision. But the opinions are change from location to location so it is very important for any decision-making process to make the best decision based on their location. This research focuses on the exploitation of Twitter for the task of sentiment analysis with respect to location. In previous research old Twitter data is taken for analyzing, training of data was based on the sentence, not the word. The research focuses on the exploitation of Twitter in real-time for the task of sentiment analysis along with the location. In this research, Pig is used for analysis as it runs standalone as well as on the Hadoop platform. And also it is best suited for both structured and unstructured data and takes less time in processing than other analyzing tools. Also, we compare the pig and hive tool.

4. PROPOSED METHODOLOGY

In this research, we fetch real-time tweets and stored theminto HDFS and then we pre-process the data using the MapReduce framework in which we can create a map task and then create variable tweets and stored all the fetch data into tweets variable. Then we can take tweets to text from data and splits the text string into an array of words by appending a delimiter between each word. Then we can explore the array of the word into vertical view and then we can perform matching between word and dictionary word and merge them by which we can get the average polarity of tweets. After that, we can classify the tweets based on their average rating.

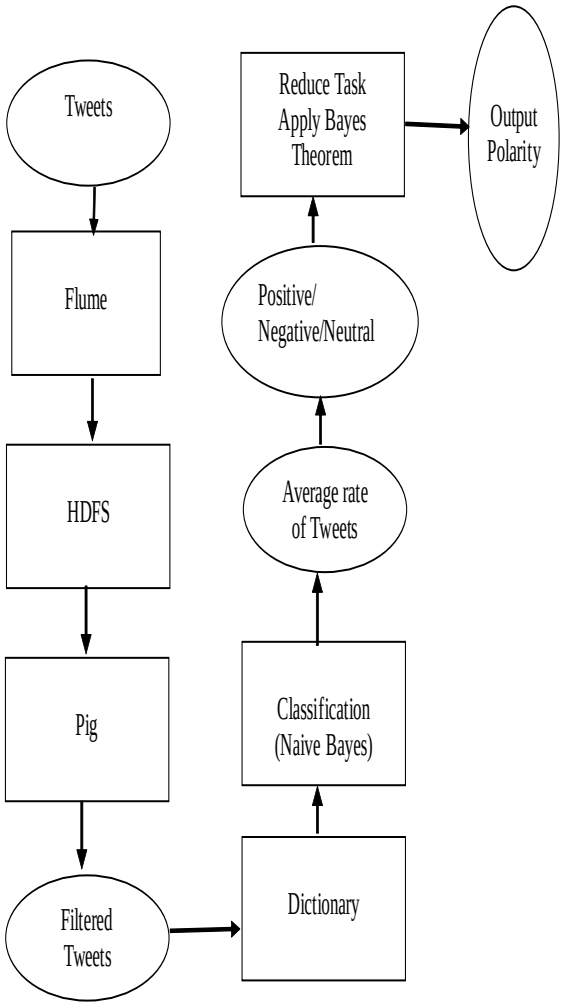


Fig.1:Complete Implementation Cycle

Algorithm 1 Get Opinion as Positive, Negative and Neutral

Input: Live tweets in textual form.

Output: Classification, Polarity of a tweet.

Retrieve tweets from Twitter API

Pre-processing and cleaning of data.

T->Tweets (t1,t2,t3…….tn)

W->Words retrieved from tweets.

D->Dictionary

Create map task,

Create a variable tweet,

Store tweets (t1,t2,t3,……tn) into tweet

For each tweet,

do{

Split tweets into array of words(w1,w2,w3…wn)

Append “”at the end of words,

Explode(words), //to provide lateral view

For each word(w),

do{

Compare (W,D) //compare tweet word with dictionary

Classify sample as Positive

If P(ω=+| x=[+1, +5])>P(ω= -| x=[-5, -1]) //ω=+,-

else classify sample as Negative

If P(ω=+| x=[+1, +5])+P(ω= -| x=[-5, -1])=0

classify sample as Neutral

}

Apply, Bayes theorem

P(w/T)=P(w).P(T/w)/P(T)

Where w is sentiment word, T is a Twitter message.

1.14Compare probabilities P(positive/T) and P(negative/T),

1.15 Generate output polarity R,

2 Create reduce task,

R= P(positive/T)-P(negative/T) //R=Resultant polarity

R =P(positive).P(T/positive)-P(negative).P(T/negative)

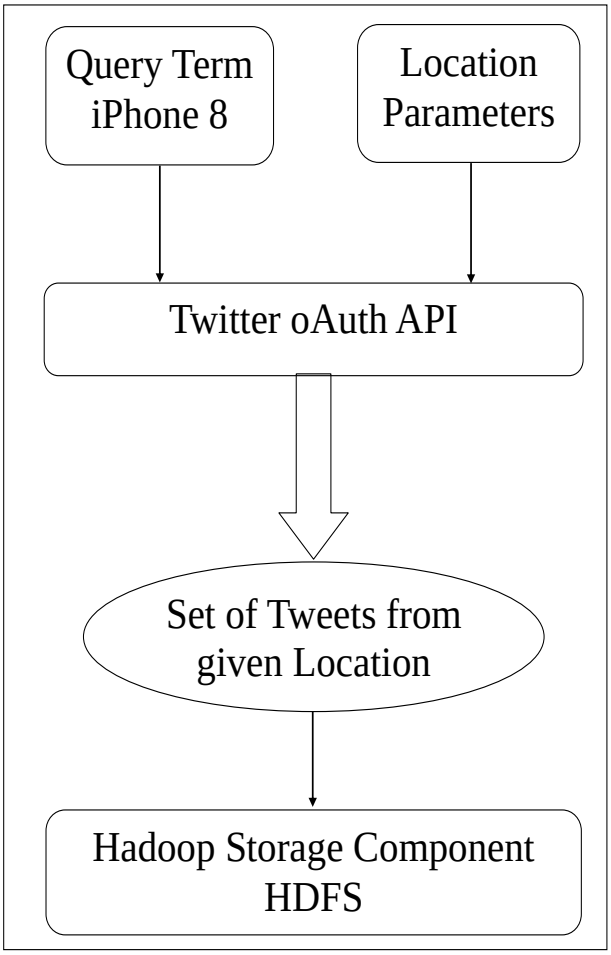
If(R>0) then predict positive opinion,

If(R<0) then predict negative opinion,

If(R=0) then predict neutral opinion,

4.1 Data Extraction

Twitter tweets were used as a data source. It is probable to excerpt tweets on a large scale from Twitter using the Twitter4j API. In our work, we used the “twitteroauth” version of the public API. This version has been implemented in Pig and can be run directly on the local host or on web servers. Once the query has been written it can be run by the API and all related twitter data will be provided as output in the browser. This data was directly inserted into an HDFS using Flume for use later on. Each record that is acquired contains several types of information like tweet id, username, text, etc. But out of that information, only the tweet id and text were useful to us. Initially, the Twitter4j API allowed tweet locations in the form of latitude and longitude to be available with every tweet where the user has made his/her location public. For our research, we decided to focus on popular countries all over the world. We extracted tweets from seven major cities all over the world. The selection of location is very limited mainly due to data availability, and language constraints. We decided to go with data from Kolkata, Cairo, Quito, Santiago, Madrid, Ackland, Athens, Arizona, and Vienna for the experiments. Each major city has a city center and the latitude and longitude that were used to define the city itself. The latitude, longitude, and radius are all values assigned to the ‘locations’ parameter in the query builder. So now we have multiple data sets each obtained from a different city. The product that we chose to analyze was the iPhone 8. Even though it is possible to analyze any product’s popularity using the defined method, the availability of data was an important issue. A reasonable amount of data about this device was available. So only the tweets which contained the term ‘iPhone 8’ in them were obtained. As we also decided to determine which feature of the iPhone 8 was most or least popular the query was enhanced using a few keywords to obtain feature specific tweets. An example would be the ‘iPhone 8 camera’. This query factors will cause the API to return only tweets that contain both iPhone 8 and camera terms together which results in tweets about the battery performance of the iPhone 8. For each tweet, the username, tweet text, location were extracted. Tweets are stored in Hadoop storage component HDFS using Flume.



Data Extraction Process

Fig.2: Data extraction procedure

4.2 Data Pre-processing

The data obtained from the API obviously contains a lot of non-relevant data. Very basic and rudimentary cleanup was performed using MapReduce. Arbitrary characters and other useless information in a tweet were filtered out before further analysis. In order to filter out these useless data we mainly used the Stanford Natural Language Processing tool by The Stanford NLP Group (SNLP Group 2015) which is an open-source natural language processing tool developed by Stanford University.

4.3 Classification

In this phase, data is classified as positive, negative, or neutral. There are many classification algorithms like Support Vector Machine Algorithm, Bayes Algorithm, Naive Bayes Algorithm, etc. In our work, we use the Naive Bayes Algorithm for classifications due to its speed and simplicity. A Naive Bayes classifier could be a straightforward probabilistic classifier supported by applying theorem (from theorem statistics) with strong (naive) independence assumptions.

Naive Bayes Classification Algorithm

Define the following symbols:

P(W | T) is the probability of class W given that we have observedT Bayesian classifiers use Bayes Theorem, which is described below

P(W | T) where is a probability of instance T being in class W

P(T | W) is a probability of generating instance T given class W

P(W) is a probability of occurrence of class W

P(T) is a probability of instance T occurring

In order to classify T's opinion as positive and negative, the probabilities are compared and the larger probability event indicates that class sentiment is more likely to happen.

According to our proposed work, Bayes theorem can be applied as:

P(w | T)=P(w).P(T | w)/ P(T)

Where w is sentiment word, T is a Twitter message.

The decision rule can be defined as

Classify sample as Positive if P **(ω=+ | x=[+1, +5]) > P(ω= - | x=[-5, -1])**

Classify sample as Negative if P **ω=+ | x=[+1, +5]) < P(ω= - | x=[-5, -1])**

Classify sample as Neutral if **P (ω= - | x=[-5, -1])=0**

Table 1.Normalization model

Sentiment score range	Assigned sentiment
Score < 0 to -5	Negative
Score = 0	Neutral
Score > 0 to +5	Positive

5. EXPERIMENTAL FINDING

5.1 Fetching Information from Twitter

All programs that attempt to hook up with Twitter and use Twitter information can all be outlined as a "Twitter APP", therefore is our Flume Agent. So first things first, we want to line up our Twitter APP.

- Go to http://apps.twitter.com and go to "Create New APP"
- Fill in the form and create an APP and access token.
- Take notes on "Consumer Key", "Consumer Secret", "Access token" and "Access token secret".

An rule utilized in Flume Agent

Set TwitterAgent.sources = Twitter

Set TwitterAgent.channel = MemChannel

Set Twitter.Agent.Sink = HDFS

Configure TwitterAgent.Source.Twitter.Type = Twitter API

Source

Configure TwitterAgent.Source.Twitter.Keys

Configure All consumer and Secret key

Configure keywords on which information is detected.

Configure HDFS location where information is hold on.

After executing these twitter agent, the flume agent established a connection between a source and sink and start fetching data and stored it into the HDFS.

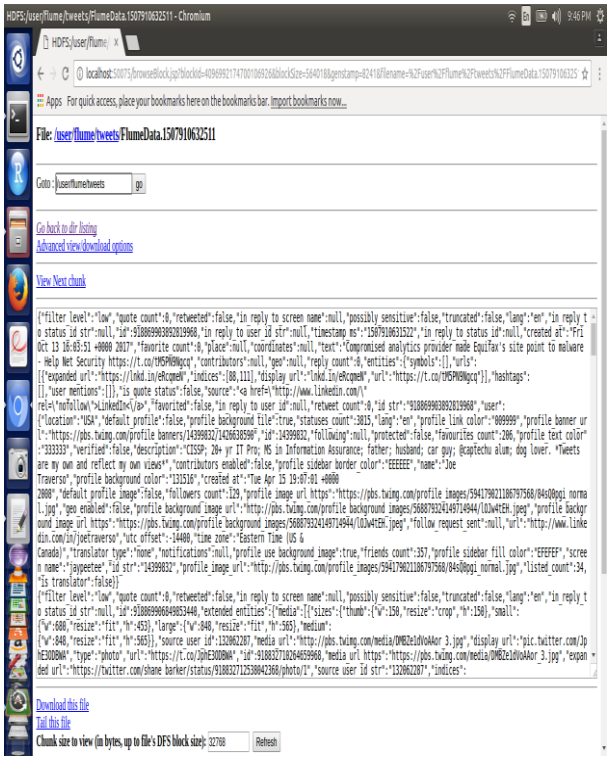


Fig.3: Twitter Data Stored into HDFS

5.2 Analyzing using Apache Pig

From this information initial we will load the data into pig wherever the filtered information wants to line up into a formatted structured specified by that we will say clearly that we've got reformed the unorganized information into a organized format. For this, we have a tendency to use some Pig JsonLoader ideas. These ideas are nothing, however, we tend to are attending to scan the information that's within the variety of JSON format for that we tend to are exploitation the elephant Json for JSON so pig will scan the JSON data and might produce a variable in our prescribed format the information. For finding polarity we are using a dictionary bases approach through which each word assigns a polarity value and after grouping, we can get the complete polarity of the text. In these, we can also take a location parameter to find the polarity based on location, from which location whats polarity value is getting.

An Algorithm used in Pig

1. Enter into the Grunt shell using command : Pig
2. A = Load the data set using com.twitter.elephantbird.pig.load.JsonLoader AS myMap ;
3. B = For each A generate myMap#'user' as User,myMap#'id' as id,myMap#'text' as text;
4. C = For each B generate User#'time_zone' as tz, id, text.
5. D = foreach C generate FLATTEN(tz) as timezone,id,FLATTEN(TOKENIZE(text)) As word;
6. Dic = Load dictionary data set using pig storage
7. E = join D by word left outer, Dic by word using 'replicated';
8. F = foreach E generate FLATTEN(rating.time_zone) as place, AVG(rating.rate) as tweet_rating;
9. G = group F by place;
10. fin = foreach G generate group,AVG(avg_rate.tweet_rating);
11. Store Output

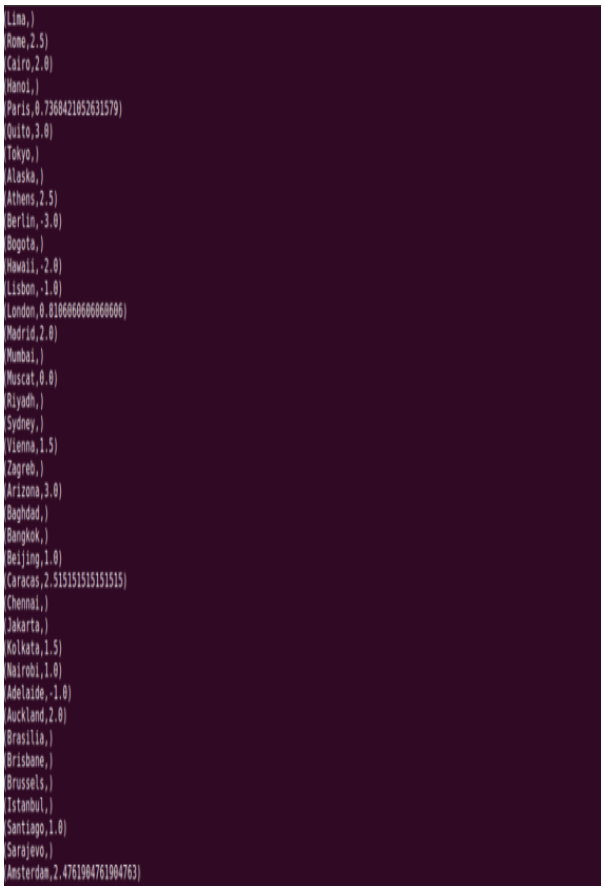


Fig.4: A sentiment score along with the location

Fig.5: Location-based positive score

5.3 Analysis using Apache Hive

We can examine the Twitter data by exploitation hive also, for these we can first load the JSON serde properties to validate the data, and convert the unstructured data into a structured form and stored it into the table. After storing it into the table we can filter the data and by using a dictionary we can get the polarity of the tweets.

An Algorithm used in Hive

1. Enter into the Hive shell using the command: hive
2. Add jar hive-serdes-1.0-SNAPSHOT.jar
3. Create table tweets(id, retweeted_status.text) ROW FORMAT SERDE 'com.cloudera.hive.serde.JSONSerDe';
4. Load data into table tweets;
5. Create table words as select id as id,split(text,' ') as words from tweets;
6. Create table wordtable as select id as id,word from words LATERAL VIEW explode(words) w as word;
7. Create table dictionary(word string,rating int) ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t';
8. LOAD DATA into TABLE dictionary;
9. Create table jointable as select wordtable.id,wordtable.word,dictionary.rating from wordtable LEFT OUTER JOIN dictionary ON(wordtable.word =dictionary.word);
10. Create table result as select id,AVG(rating) as rating from jointable GROUP BY id order by rating DESC;

5.4 Comparison between Pig and Hive

We can accomplish the same analysis on Twitter data by exploitation of Apache Hive and it is well known that both Pig and Hive demonstrate similar results. Apache Pig render more control over the data even if the data is unorganized, whereas Hive is efficient in handling structured data. For our Twitter datasets, we can analyze the data by using hive and pig on fewer instance data and we say that pig has finer efficiency as compared to the hive on processing JSON data.

Table 2z:Execution time is taken by Hive and Pig

Time taken by in sec	Hive	Pig
10 MB	19	16
15 MB	30	25
20 MB	44	34

.

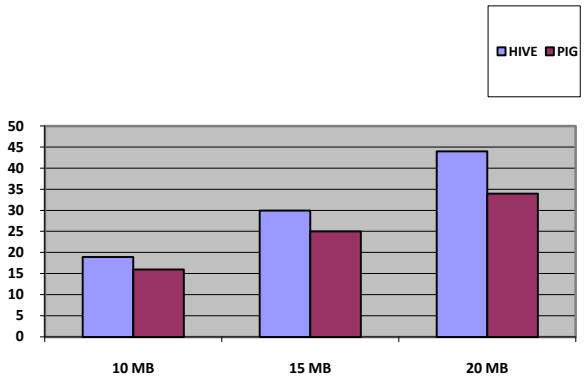


Fig. 6: Execution time is taken by Pig and Hive

6. CONCLUSION

In this research, we discussed a methodology by which it is possible to determine the popularity/opinion/sentiment of a product in different locations. For our analysis, we chose the iPhone 8 as a reasonable amount of tweets based on the iPhone 8 was available. The number of tweets must be significant for accurate results. Therefore, even if a good does not have a large number of tweets at any given moment, we could accumulate tweets over a period of several weeks or months. For the choice of a location, we choose seven popular cities all over the world. The reason behind this is also data accessibility. But the methodology defined is much generalized and can be applied to tweets from any country for any product as long as a suitable number of tweets can be obtained. Initially, the tweets were fetched and processed using Flume and Pig. Finally, the data were presented graphically. The analysis can also be done using the Hive tool, but Pig is best suited as compare with Hive.

7. REFERENCES

[1] Ankur Goel, Jyoti Gautam, Sitesh Kumar, "Real Time Sentiment Analysis of Tweets Using Naive Bayes", 2016 2nd International Conference on Next Generation Computing Technologies (NGCT-2016) Dehradun, India 14-16 October 2016, IEEE.

[2] Mrunal Sogodekar, Shikha Pandey, Isha Tupkari, Amit Manekar, "Big Data Analytics: Hadoop And Tools", 2016 IEEE Bombay Section Symposium (IBSS), IEEE 2016.

[3] Aditya Bhardwaj, Vanraj, Ankit Kumar, Yogendra Narayan, Pawan Kumar, " Big Data Emerging

Technologies: A case-study with Analyzing Twitter Data using Apache Hive", IEEE 2015.

[4] Sagiroglu, S., & Sinanc, D., "Big data: A review", IEEE International Conference on Collaboration Technologies and Systems (CTS), 2013.

[5] Can Uzunkayaa, Tolga Ensaria, Yusuf Kavurucu, "Hadoop Ecosystem and Its Analysis on Tweets", World Conference on Technology, Innovation and Entrepreneurship, Procedia - Social and Behavioral Sciences 195 (2015) 1890 – 1897, Elsevier 2015.

[6] Manoj Kumar Danthala, "Tweet Analysis: Twitter Data processing Using Apache Hadoop", International Journal Of Core Engineering & Management (IJCEM) Volume 1, Issue 11, February 2015.

[7] Sitaram Asur, Bernardo A. Huberman, "Predicting the future with social media", International conference on Web intelligence and intelligent agent technology (WI-IAT), IEEE/WIC/ACM vol. 1, 2010.

[8] Judith Sherin Tilsha S, Shobha M.S, "A Survey on Twitter Data Analysis Techniques to Extract Public Opinion", IJARCSE, Vol. 5, Issue 11, Nov 2015.

[9] M. Trupthi, Suresh Pabboju, G. Narasimha, "Sentiment Analysis on Twitter Using Streaming API", 2017 IEEE 7th International Advance Computing Conference (IACC), 2017.

[10] M. Mazhar Rathore, Anand Paul, Awais Ahmad, Muhammad Imran, Mohsen Guizani, "Big Data Analytics of Geosocial Media for Planning and Real-Time Decisions", SAC Symposium Big Data Networking Track, IEEE ICC 2017.

[11] Nikitha Johnsirani Venkatesan, Earl Kim, Dong Ryeol Shin, "PoN: Open Source solution for Real-time Data Analysis", IEEE 2016.

[12] Divya Sehgal and Dr. Ambuj Kumar Agarwal, "Sentiment Analysis of Big Data Applications using Twitter Data with the Help of Hadoop Framework", Proceedings of the SMART -2016, IEEE Conference 5th International Conference on System Modeling & Advancement in Research Trends.

[13] Nikitha Johnsirani Venkatesan, Dong Ryeol Shin, Earl Kim, "PoN: Open Source solution for Real-time Data Analysis", IEEE, 2016.

[14] Syed Akib Anwar Hridoy, M. Tahmid Ekram, Mohammad Samiul Islam, Faysal Ahmed and Rashedur M. Rahman, "Localized twitter opinion mining using sentiment analysis", Springer open journal, 2011.