ISSN:0975-9646

Real Time Opinion Mining of Twitter Data

Narahari P Rao[#1], S Nitin Srinivas[#2] and Prashanth C M[*3]

[#] B.E, *Dept of CS&E, SCE, Bangalore, India*

[*] *Prof & HOD Dept of CS&E, SCE, Bangalore, India*

***Abstract*—Social Networking sites provides tremendous impetus for Big Data in mining people's opinion. Public API's catered by sites such as Twitter provides us with useful data for perusing writer's attitude apropos of a particular topic, product etc. To discern people's opinion, tweets are tagged into positive, negative or neutral indicators. This paper provides an effective mechanism to perform opinion mining by designing a end to end pipeline with the help of Apache Flume ,Apache HDFS, Apache Oozie and Apache Hive. To make this process near real time we study the workaround of ignoring Flume tmp files and removing default wait condition from Oozie job configuration. The underlying architecture employed here is not restricted only to opinion mining but also has a gamut of applications. This paper explores few of the use cases that can be developed into actual working models.**

Keywords—Opinion Mining, Big Data, Real time Tweet Analysis, Oozie workaround.

I. INTRODUCTION

Opinions are subjective expressions that outline people's, appraisals, feelings or sentiments toward entities, events and their properties. Recently there has been a massive escalation in use of Social Networking sites such as Twitter to express people's opinions. Impelled by this growth, companies, media, review groups are progressively seeking ways to mine Twitter for information about what people think and feel about a particular product or service. Twitter data is a valuable source of information for marketing intelligence and trend analysis in all industries. Twitter generates gigantic data that cannot be handled manually hence the requirement of automatic categorization. Tweets are unambiguous short texts messages that are up to a maximum of 140 characters. These texts are polarized based on the nature of the comment. Focus of this paper is to provide an automated mechanism for collecting, aggregating, streaming and analyzing tweets in near real time environment and a glimpse of two of its use case scenarios.

II. RELATED WORKS

Opinion mining is one of the most popular trends in today's world. Lot of research and Literature surveys are being done in this sector. Bo Pang and Lillian Lee are pioneers in this field. Current works in this field which uses a mathematical approach using algorithms for opinion polarity are based on a classifier trained using a collection of annotated text data. Before training, data is preprocessed so as to extract only the main content .Some of the classification methods have been proposed are Naïve Bayes, Support Vector Machines, K-Nearest Neighbors etc. Continuous research is being done to determine most efficient method for opinion mining.

III. OUR APPROACH

A. *Data From Twitter*

Twitter provides us with a Streaming API which will be employed to obtain a constant stream of tweets enabling us to collect and analyze user opinion. The Streaming API works by making a request for a specific type of data which is filtered by keyword, a user, geographic area etc. Once connection to the Twitter API is established via the Streaming API, data collection takes place. The tweets collected will be encoded in JavaScript Object Notation (JSON). JSON provides us with a way to encode this data. The whole tweet is regarded as a dictionary consisting of various fields. The fields may be contributors (indicates users who have authored the tweet), coordinates (Represents the geographic location of the Tweet as reported by client application), favorite_count (No. of times the tweet has been "favorited"), text (actual text of the tweet) and several other fields.

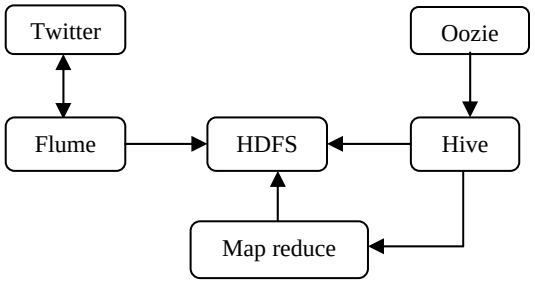

Fig. 1 Workflow

Gathering Data with Apache Flume

To automate the movement of tweets from the API to HDFS, without our manual intervention, Flume is used. Apache Flume is a reliable and distributed system for effectively gathering and moving large amounts of data from various sources to a common storage area. Major components of flume are source, memory channel and the sink.

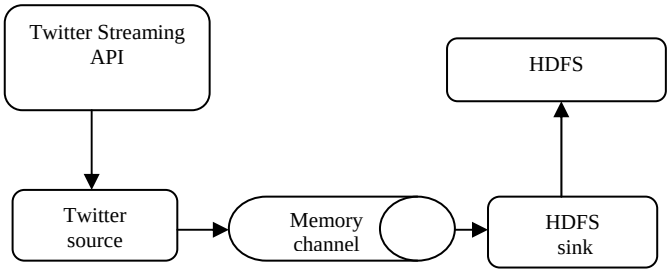

Fig. 2 Components of Flume.

Twitter source is an event-driven source that uses Twitter4j library for accessing streaming API. Tweets are collected and aggregated into fundamental units of data called as an event. An event incorporates a byte payload and an optional header. The coordination of event-flows from the streaming API to HDFS is undertaken by Agent. The acquired tweets are stored into one or more memory channels. A memory channel is a temporary storage that uses an in-memory queue to retain events until they are ingested by the sink. Using memory channel, tweets are processed in batches that can be configured to hold a constant number of tweets. To procure tweets for a given keyword filter query is used. Sink writes events to a pre configured location. This system makes use of the HDFS-sink that deposits tweets into HDFS.

B. Hadoop

Hadoop is a open source framework for processing and storing large datasets over a cluster.It is used in handling large and complex data which may be structured,unstructured or semi-structured that does not fit into tables. Twitter data falls into the category of "semi-structured" data which can be best stored and analyzed using Hadoop and it's underlying file system.

C. Hadoop Distributed File System

Hadoop Distributed File System (HDFS) is a distributed file system which rests on top of the native file system and is written in java. It is highly fault tolerant and is designed for commodity hardware. HDFS has a high throughput access to application and is suitable for applications with large amount of data. The master-server architecture of HDFS having single name node helps in regulating the file system access.

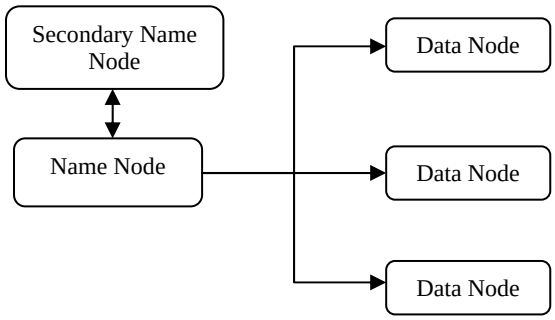

Fig. 3 Hadoop architecture

Requests from file system clients are handled by the data nodes. Data is stored as Input splits (blocks) on the underlying file system. The replication factor is set as 3 by default in order to maintain redundancy of data. In this case, huge amounts of tweets are collected, stored and analyzed.

D. MapReduce

MapReduce is one of the two major components of Hadoop. It is a programming paradigm designed to assist computing which involves data intensive operations. MapReduce comprises of two distinct jobs that Hadoop programs perform. MapReduce jobs are controlled by a software daemon known as the JobTracker. The JobTracker resides on a master node. The JobTacker initiates the map() and reduce() jobs in the data nodes where the TaskTracker daemon resides. MapReduce requires a Java programmer. Other than very basic applications, MapReduce requires multiple stages leading to cumbersome codes. It's users have to reinvent common functionalities such as joining and querying which are provided by Hive as inbuilt functions. The tweets are queried using Hive whose Execution engine in turn generates Map and Reduce jobs in order to query out the parts of the tweets which the user is interested in.

E. Hive

After congregating the tweets into HDFS they are analyzed by queries using Hive. Apache Hive data warehouse software facilitates querying and managing large datasets residing in distributed storage. Hive provides a mechanism to project structure onto this data and query the data using a SQL-like language called HiveQL. In opinion mining system, hive is used to query out interested part of the tweets which can be an opinion, comments related to a specific topic or a trending hash tag. Twitter API loads the HDFS with tweets which are represented as JSON blobs. Processing twitter data in relational database such as SQL requires significant transformations due to nested data structures. Hive facilitates an interface that provides easy access to tweets using HiveQL that supports nested data structures. Hive compiler converts the HiveQL queries into map reduce jobs. Partition feature in hive allows tweet tables to split into different directories. By constructing queries that includes partitions, hive can determine the partition comprising the result. The location of twitter tables are explicitly specified in "Hive External Table" which are partitioned. Hive uses SerDe (Serializer-Deserializer) interface in determining record processing steps. Deserializer interface intakes string of tweets and translates it into a Java Object that Hive can manipulate on. The Serializer interface intakes a java Object that Hive has worked on and converts it into required data to be written on HDFS.

F. Oozie

Once the Tweets are loaded into HDFS, it is staged for querying by creating an external table that is partitioned in Hive. The continuous stream of tweets from the Twitter API will create new files that needs to be partitioned regularly after a specific time interval .To automate the periodic process of adding partitions to our table as the new data comes in, Apache Oozie is used. Oozie runs workflow jobs along with actions that in turn run Hadoop jobs. A workflow comprises of actions arranged in a control dependency Direct Acyclic Graph (DAG). Control dependency from one action to another denotes that the second action can run only after the completion of the first action. The Coordination-application job of Oozie schedules the execution of recurring hive queries, adding new twitter data to Hive table every hour.

IV. OOZIE HURDLE

The configuration of the Oozie workflow will only add a new partition to the Hive table when a new hour rolls forward. Thus there is a wait event built-into the Oozie workflow. The wait condition occurs due to a locked tmp file generated by Flume. Due to this locked file Hive is restricted from querying the Hive external table while the tmp file is being written. This wait event forbids the process from being real time.

V. ENHANCEMENT

To eliminate the wait condition and to make the process work in real-time following steps can be employed:

a) Create a custom package to ignore Flume temp files:

```
public class FileFilterExcludeTmpFiles implements PathFilter
 {
        public boolean accept(Path  p)
        {
                String name = p.getName();
                return  !name.endsWith(".tmp");
                //This line ignores the files ending with .tmp
        }
}
```

*b) Removing Oozie wait condition:*The Oozie workflow by default has defined a readyIndicator that acts as a wait event. It directs the workflow to create a new partition only after a period of one hour.By removing the readyIndicator the wait condition is overlooked.

c) The Hive Configuration File is modified to indicate the location of the new Java class created.

VI. RESULTS

After streaming the tweets into HDFS in real time, hive is used in analyzing the tweets. Tweets are tagged as documents where categories are the hash tags defined in the Flume configuration file. Later the tweets are grouped as positive, negative and neutral based on subjectivity corpus forming a dictionary of words and its polarity.

The sample example shown in Fig 4 is obtained by mining for tweets with hash tag "bigdata".

TABLE I.

OPINION	COUNT
POSITIVE	729
NEGATIVE	150
NEUTRAL	665

Fig. 4 Sample of Tabular result

Data Visualization

The sample output can be visualized by bubble plot analysis using data visualization tool such as knime.

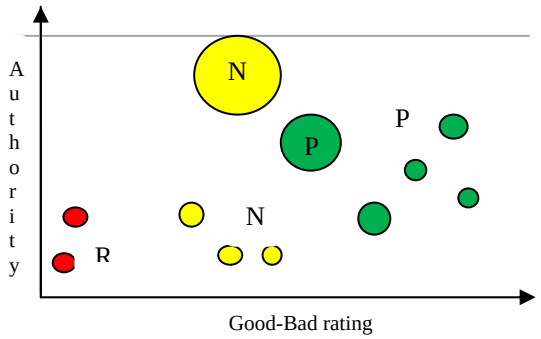

Fig. 5 Bubble plot

N-neutral, P-positive, R-negative

Good-Bad rating attribute indicates the overall polarity of tweets. Higher the authority attributes higher prominence for the rating.

VII. TIME EFFICIENCY

Time efficiency is an important feature of this project. Lower response time is achieved by use of custom Java package created. This reduces the overhead time of one hour for each data set. Also the use of Hadoop ensures the distributed processing and it also lowers the access time. Latency in fig.6 denotes the overhead time for accessing Hive external tables for querying and analyzing. The overhead time of one hour is reduced significantly to few minutes. Hence overall the time efficiency increases owing to the above mentioned factors.

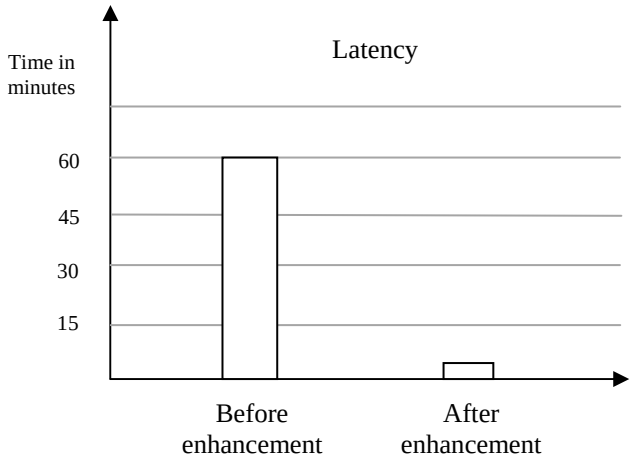

Fig. 6 Latency graph

X. APPLICATION

ASSISTING COLLABORATIVE DECISION MAKING:

Collaborative decision-making is a situation faced when individuals collectively make a choice from the alternatives before them. Real Time Opinion Mining System can be used as a platform for making group decisions by taking into account the opinions of individuals in real time.

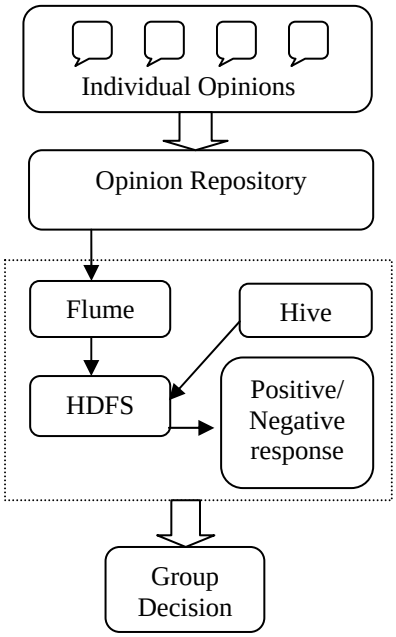

Fig. 7 Use case workflow

The individual opinions are aggregated together in opinion repository and are streamed into HDFS by Flume. Using HiveQL, opinions are classified into positive and negative responses which assist group decision making.

CORRUPTION VIGILANCE SYSTEM

Corruption is a serious issue adversely affecting the Indian economy. In the year 2014, India was ranked 85th out of 175 countries in the Corruption Perception Index. The biggest problem of Corruption is that it is not being exposed as and when it happens. As a result of which the offenders evade punishment with ease. We propose a system which uses tweets from users to expose corruption in real time. The underlying architecture of this application can be visualized as shown in the Fig. 8.The workflow scheme depicts a scenario where a person witnessing an act of corruption tweets with a unique hash tag comprising the details of the offender. The tweet is then procured into the HDFS employing Flume and streaming API in real time. The collected tweet is queried for details such as the name of the offender, location and is stored in HDFS. This completes the tweet collection phase. The offender's identity queried, stored in the HDFS is now validated for authenticity to detect pseudonymous complaints concluding verification phase in the work flow. In the final phase, a table holding details of the incident as tweeted by the user is stored and disclosed to the public. Once developed into a working model, this system can inspire a social movement involving public to provide a helping hand for the government in curtailing acts of corruption.

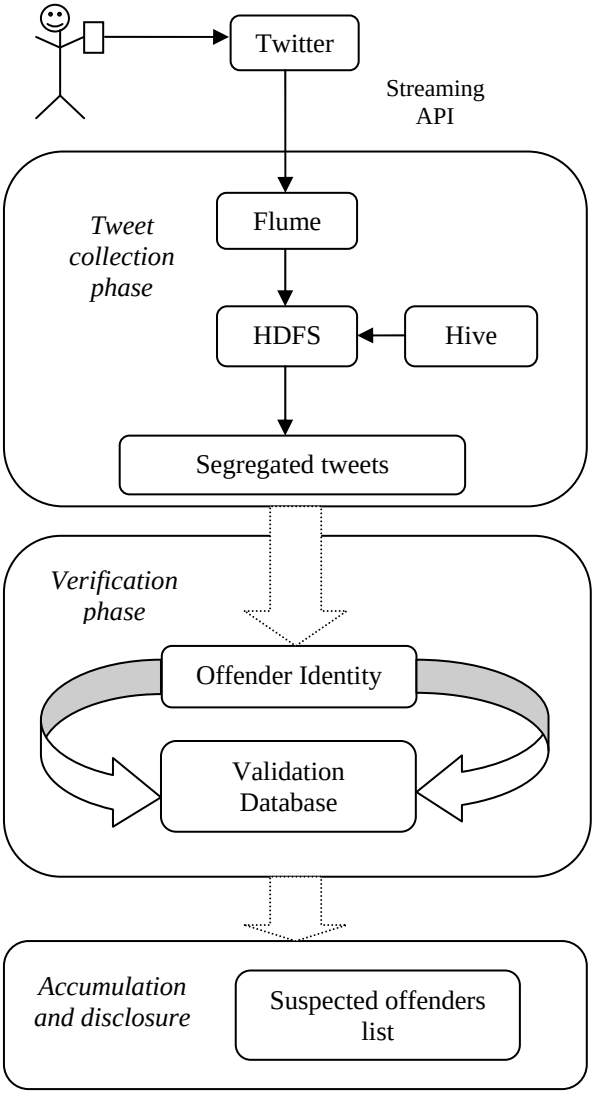

Fig. 8 Use case workflow

XI. SCOPE FOR FUTURE WORK

A. Deployment in Cloud: It is an arduous task for average system to process, analyze and store huge amount of data sets. Hence there is a requirement of powerful machines which are made available through cloud platforms (IAAS or VPC).Deploying the whole system in cloud provides hassle-free access to it.

B. Secure e-Voting System: A secure mechanism is required for expressing people's consensus on policy initiatives and electoral procedures in countries that follow direct democracy.

Direct democracy involves people's opinion in government decision making by conducting regular referendums involving people casting their votes in polling booths. These referendums require physical presence of voters in polling stations in frequent intervals causing disruption in their daily routines. Whereas a secure model having similar architecture can provide a mechanism to cast people's vote from their home.

XII. CONCLUSION

Opinion Mining is a very wide branch for research. We have covered some of its important aspects. The same architecture could be used for a variety of applications designed to look at Twitter data, such as identifying spam accounts, or identifying clusters of keywords. Taking the system even further, the general architecture can also be expanded to other social media platform usages like Facebook, movie reviews, personal blogs, etc. Evidently, taking into account all the constraints, this method is one of the most efficient ways to perform opinion mining in real time.

REFERENCES

[1] Apporv Agarwal, Jasneet Singh Sabarwal, *"End to End Sentiment Analysis of Twitter Data"*

[2] A. Pak and P. Parouek, *"Twitter as a corpus for sentiment analysis and opinion mining,"* in Proceedings of LREC, vol. 2010.

[3] *"Sentimental Analysis",* Inc. [Online]. Available: http://www.cs.uic.edu/~liub/FBS/sentiment-analysis

[4] CDH-googlegroup *https://groups.google.com/a/cloudera.org/forum/?fromgroups=#!topic/cdh-user/FWH80lehYxk*

[5] T. White, "*The Hadoop Distributed Filesystem,*" Hadoop: The Definitive Guide, pp. 41-73, GravensteinHighwaNorth, Sebastopol: O'Reilly Media, Inc., 2010.

[6] Theresa Wilson, Joanna Moore, Efthymios Kouloumpis,*"Twitter Sentiment Analysis – The Good, the Bad and the OMG"*

[7] Knime_Socialmedia_whitepaper – *https://tech.knime.org/files/knime_social_media_white_paper.pdf*