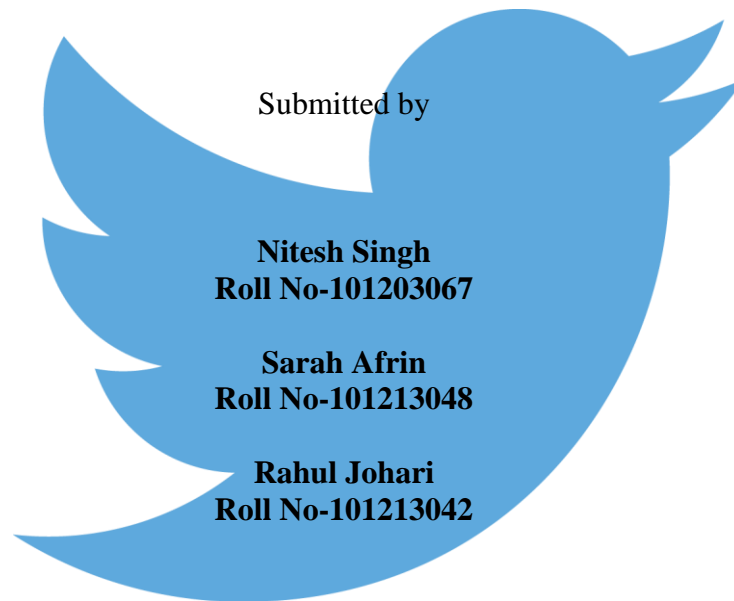


PROJECT REPORT
(PROJECT SEMESTER TRAINING)

TWITTER ANALYSIS



Under the Guidance of

Venkat Billa Sir (Big Data Faculty at HP)

Department of Computer Science and Engineering
THAPAR UNIVERSITY, PATIALA

August 2014

DECLARATION

I hereby declare that the project work entitled Twitter Analysis is an authentic record of my own work carried out at HP as requirements of 6 weeks summer training for the award of degree of B.E. (Computer Science & Engineering), Thapar University, Patiala, under the guidance of Venkat Sir, during 16 June to 31 July, 2014.

I further declare that no part of this report is copied from Internet or any other source.

(Signature of student)
Nitesh Singh
101203067

(Signature of student)
Sarah Afrin
101213048

(Signature of student)
Rahul Johari
101213042

Date: _____

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere gratitude to H.P.E.S NOIDA to have recruited me to undergo my six weeks project training here. My project training at H.P.E.S NOIDA has left me with an enriching experience, equipping me with superior technical and inter-personal skills, providing me a congenial environment to apply my full creative potential and encouraging me to pursue my ideas and goals. I would like to express my earnest thanks to the fine people around me, who helped me in completing this project work.. Firstly, I would like to thank Mr.Venkat Billa sir for giving me a golden opportunity to work in this highly interesting and significantly important project Twitter Analysis. Their continued support, guidance and vision has not only helped me in this project but enlightened me on various aspects which will be invaluable for my future. Also I wish to express my heartfelt gratitude towards my professors at Thapar University who taught the fundamental essentials, honed my skills and made me ready to face the challenges in corporate life. Without their valuable guidance it would have been extremely difficult to grasp and visualize the project

Nitesh Singh
101203067

Sarah Afrin
101213048

Rahul Johari
101213042

ABSTRACT

Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process using on-hand data management tools or traditional data processing applications.

Social media has become a major platform for information sharing.

Our key focus for analysis. It helps your organization answer the following types of questions:

What are consumers saying and hearing about my brand?

What are the most talked about product attributes in my product category? Is the feedback good or bad?

What is the competition doing to excite the market?

What is the reputation of the new vendors I am considering?

Due to its openness in sharing data, *Twitter* is a prime example of social media in which researchers can verify their hypotheses, and practitioners can mine interesting patterns. In our project we do Twitter Data Analysis to Find Sentiment and other useful information using **Apache Hadoop**.

Sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state (that is to say, the emotional state of the author when writing), or the intended emotional communication (that is to say, the emotional effect the author wishes to have on the reader).

Apache Hadoop is an open-source software framework for storage and large-scale processing of data-set (Structured and Unstructured). Hadoop is an Apache top-level project being built and used by a global community of contributors and users. It is licensed under the Apache License 2.0.

COMPANY PROFILE

HP the largest IT company in the world, through its partner offers high quality Courses for 4 & 6 weeks. Few Courses in Summer Training Program are Programming Techniques using 'C', Network Management & Security, PHP, VLSI, VHDL & PCB Design, Data Structure using C Language, Core Java, J2EE-with Struts Framework, J2EE-Struts with Hibernate Framework, Android, ASP.NET With C#, Embedded & Robotics-Basics, Embedded & Robotics-Advanced, ARM, Linux Administration with Scripting, Project Management, IT Operations and Services Workshop, Advance Concepts of Networking.

With ever changing technologies & methodologies, the competition today is much greater than ever before. The industrial scenario needs constant technical enhancements to cater to the rapid demands. If we are an engineering student or pursuing graduate/post-graduate level IT degree then you may have already heard the term “Summer Training”. These trainings are important because it is the best way to acquire and clear our concepts about our respective fields.

TABLE OF CONTENTS

S.No.	TOPIC	PAGE No.
1.	Declaration	(i)
2.	Acknowledgements	(ii)
3.	Abstract	(iii)
4.	Company Profile	(iv)
5.	Introduction	
6.	Background	
7.	Work Flow	
	i. Analysis using Hive	
	ii. Analysis using Map Reduce	
7.	Limitations	
8.	Conclusion	
9.	Bibliography	

INTRODUCTION

About of the project:

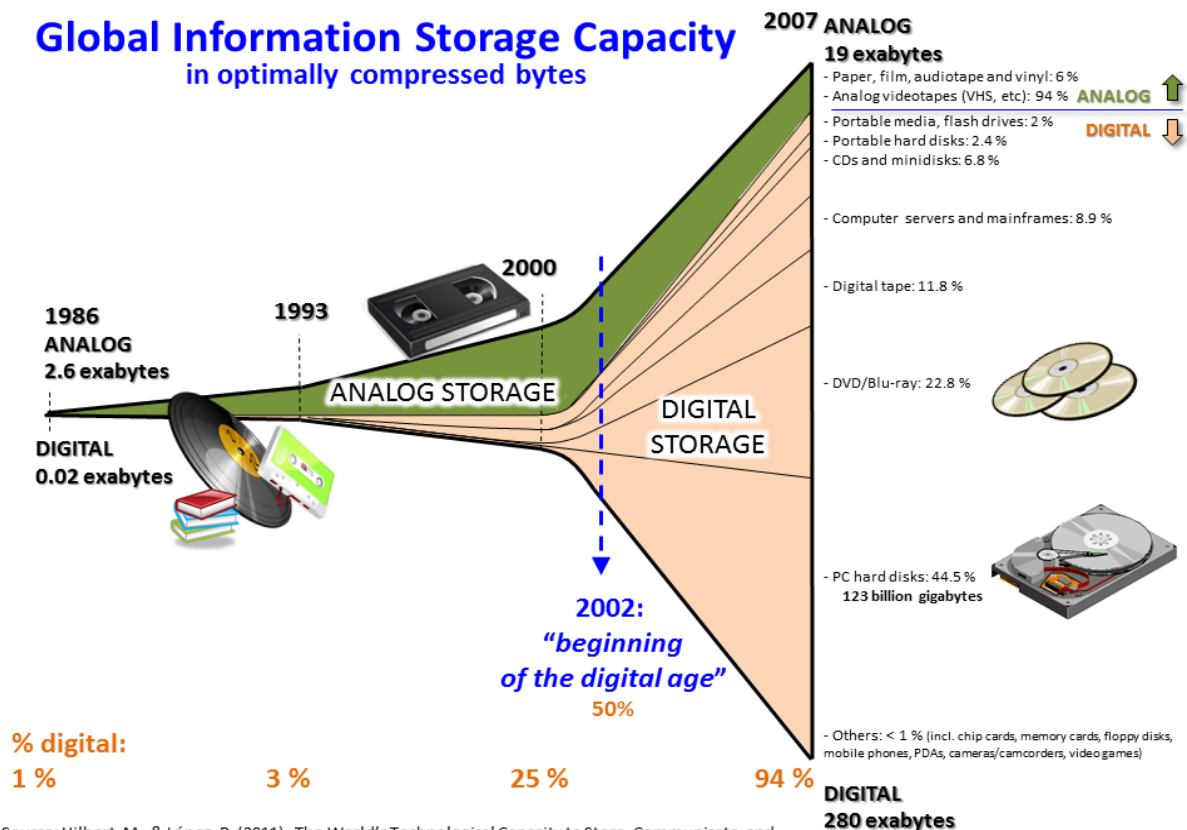
Social networking sites nowadays are contributing a lot towards big data. In order to find the interesting patterns or trends from this huge data, data scientists need to clean, integrate, aggregate and analyze the data. The purpose of this project is to find out trends by aggregating the data in social networking site such as Twitter.

Analysis of Twitter Data to evaluate sentiment to the user tweets, find the hash-tag used by the users and to find the most popular user for the twitter data related to

- ☐ *Commonwealth Games*
- ☐ *Eid Festival*
- ☐ *India Vs England Test Match*
- ☐ *Gaza attack*
- ☐ *Launch of Xiaomi Smartphone on Flipkart*

Every day, we *create 2.5 quintillion* bytes of data (source:IBM) — so much that 90% of the data in the world today has been created in the last two years alone. This data comes from everywhere: sensors used to gather climate information, posts to social media sites, digital pictures and videos, purchase transaction records, and cell phone GPS signals to name a few. This data is **big data**.

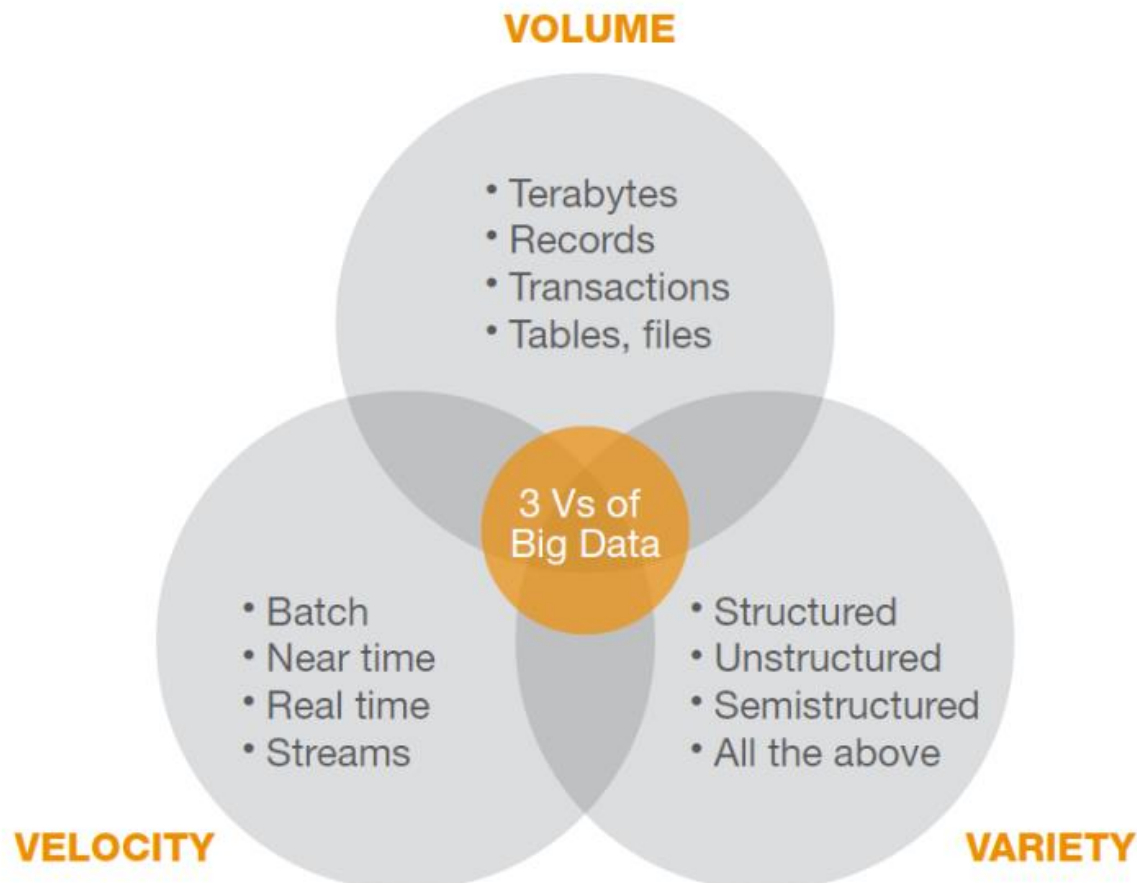
What is Big Data?



Source: Hilbert, M., & López, P. (2011). The World's Technological Capacity to Store, Communicate, and Compute Information. *Science*, 332(6025), 60–65. <http://www.martinhilbert.net/WorldInfoCapacity.html>

Big data usually includes data sets with sizes beyond the ability of commonly used software tools to capture, curate, manage, and process the data within a tolerable elapsed time. Big data sizes are a constantly moving target, as of 2012 ranging from a few dozen terabytes to many petabytes of data in a single data set.

In a 2001 research report and related lectures, META Group analyst Doug Laney defined data growth challenges and opportunities as being three-dimensional, i.e. increasing volume (amount of data), velocity (speed of data in and out), and variety (range of data types and sources). We call this as "3Vs" model for describing big data.



In 2012, Gartner updated its definition as follows: "Big data is high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization".

The growing maturity of the concept gives us a more sound difference between big data and Business Intelligence, regarding data and their use:

Business Intelligence uses descriptive statistics with data with high information density to measure things, detect trends etc.

Big data uses inductive statistics and concepts from nonlinear system identification to infer laws (regressions, nonlinear relationships, and causal effects) from large data sets to reveal relationships, dependencies and perform predictions of outcomes and behaviours.

Big data can also be defined as "Big data is a large volume unstructured data which cannot be handled by standard database management systems like DBMS, RDBMS or ORDBMS".

2. Project objectives:

The core objective of the project is to understand the components and core technologies related to content retrieval, storage and data intensive analysis of large corpus of data collected over a specific period of time. One of the important objectives is to demonstrate the analysis using visualization tools.

Content Retrieval: The large amount of data is collected using java Twitter streaming API.

Data Processing: Data collected over a period of time is processed by using java and distributed processing software framework developed by Apache Hadoop and using map reduce programming model and Apache hive frame work.

Storage: This data is stored in a certain format (HDFS: Hadoop Distributed File system) so as to form key value pair which is needed to feed to mapper in map-reduce programming approach. The data is stored in Hadoop2 Distributed File System.

Data Analysis: The output obtained from reducer phase is analysed.

Visualization: Various ongoing trends on social networking sites are aesthetically represented using Google Visualization Tools.

3. Project Approach:

1. Studying Hadoop 1.1 architecture
2. Configuring Hadoop on the machine.
The installation guide is:
http://j.mp/install_instruction_hadoop
3. Understanding map reduce functionality
4. Implementing simple word count program in Map Reduce
5. Studying Twitter API
6. Understanding hive Framework
7. Analysis Data using hive
8. Data Visualisation
9. Interactive Data Presentation

BACKGROUND

Big data analytics is the use of advanced analytic techniques against very large, diverse data sets that include different types such as structured/unstructured and streaming/batch, and different sizes from terabytes to zettabytes.

Big data analytics refers to the process of collecting, organizing and analyzing large sets of data (big data) to discover patterns and other useful information. Not only will big data analytics help you to understand the information contained within the data, but it will also help identify the data that is most important to the business and future business decisions. Big data analysts basically want the knowledge that comes from analyzing the data.

APACHE HADOOP

Apache Hadoop™ was born out of a need to process an avalanche of Big Data. The web was generating more and more information on a daily basis, and it was becoming very difficult to index over one billion pages of content. In order to cope, Google invented a new style of data processing known as MapReduce. A year after Google published a white paper describing the MapReduce framework, Doug Cutting and Mike Cafarella, inspired by the white paper, created Hadoop to apply these concepts to an open-source software framework to support distribution for the Nutch search engine project. Given the original case, Hadoop was designed with a simple write-once storage infrastructure.

Hadoop has moved far beyond its beginnings in web indexing and is now used in many industries for a huge variety of tasks that all share the common theme of lots of variety, volume and velocity of data – both structured and unstructured. It is now widely used across industries, including finance, media and entertainment, government, healthcare, information services, retail, and other industries with Big Data requirements but the limitations of the original storage infrastructure remain.

It is a framework of many components put together but the two main basic components are:

1. Computation component: The computation tier uses a framework called MapReduce.
2. Distributed storage component: A distributed file system called HDFS provides storage

HDFS

Hadoop Distributed File System: HDFS, the storage layer of Hadoop, is a distributed, scalable, Java-based file system adept at storing large volumes of unstructured data. It runs on existing file system.

MapReduce

MapReduce: MapReduce is a software framework that serves as the compute layer of Hadoop. MapReduce jobs are divided into two (obviously named) parts. The “Map” function divides a query into multiple parts and processes data at the node level. The “Reduce” function aggregates the results of the “Map” function to determine the “answer” to the query.

Map reduce programs are used to compute results from the data saved in hdfs.

The important innovation of MapReduce is the ability to take a query over a dataset, divide it, and run it in parallel over multiple nodes. Distributing the computation solves the issue of data too large to fit onto a single machine. Combine this technique with commodity Linux servers and you have a cost-effective alternative to massive computing arrays.

At its core, Hadoop is an open source MapReduce implementation. Funded by Yahoo, it emerged in 2006 and, according to its creator Doug Cutting, reached “web scale” capability in early 2008.

Programming Hadoop at the MapReduce level is a case of working with the Java APIs, and manually loading data files into HDFS.

Working directly with Java APIs can be tedious and error prone. It also restricts usage of Hadoop to Java programmers. Hadoop offers two solutions for making Hadoop programming easier Pig and Hive.

Hive

Hive is a Hadoop-based data warehousing-like framework originally developed by Facebook. It allows users to write queries in a SQL-like language called HiveQL, which are then *converted to MapReduce*. This allows SQL programmers with no MapReduce experience to use the warehouse and makes it easier to integrate with business intelligence and visualization tools such as Microstrategy, Tableau, Revolutions Analytics, etc.

Hive enables Hadoop to operate as a data warehouse. It superimposes structure on data in HDFS and then permits queries over the data using a familiar SQL-like syntax.

Other components in hadoop framework are

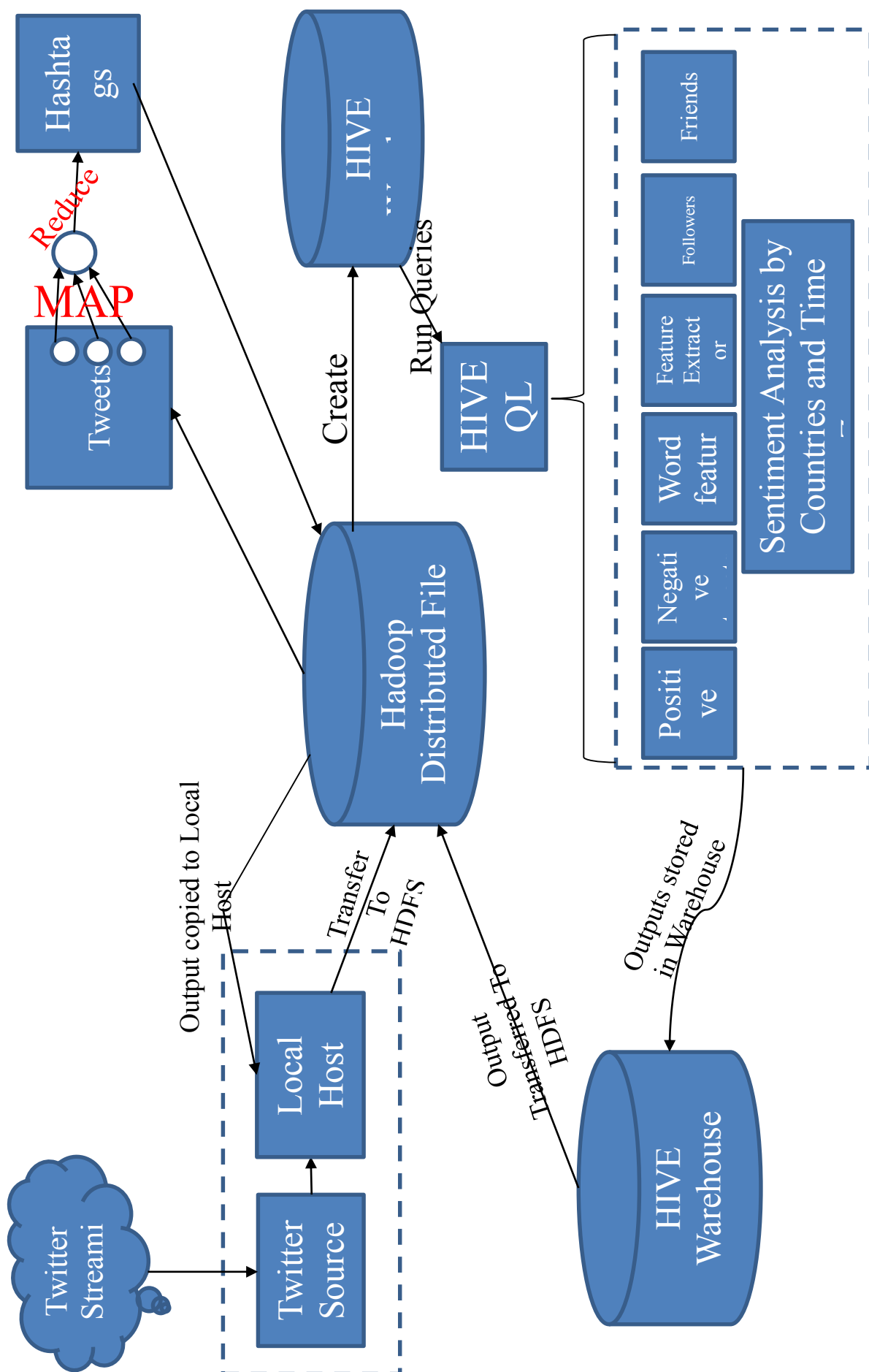
- | | |
|------------------------------------|---|
| <input type="checkbox"/> Flume | Collection and import of log and event data |
| <input type="checkbox"/> HBase | Column-oriented database scaling to billions of rows |
| <input type="checkbox"/> HDFS | Distributed redundant file system for Hadoop |
| <input type="checkbox"/> Hive | Data warehouse with SQL-like access |
| <input type="checkbox"/> Mahout | Library of machine learning and data mining algorithms |
| <input type="checkbox"/> MapReduce | Parallel computation on server clusters |
| <input type="checkbox"/> Pig | High-level programming language for Hadoop computations |
| <input type="checkbox"/> Oozie | Orchestration and workflow management |
| <input type="checkbox"/> Sqoop | Imports data from relational databases |
| <input type="checkbox"/> Zookeeper | Configuration management and coordination |

Each of these was developed to address a gap in Hadoop: Hive, Impala and HBase to make Hadoop look something like a database; Pig to lower the cost of developing MapReduce programs; and, Mahout to allow programmers to avoid re-inventing statistical algorithms every time they author a new MapReduce program.

Why Hadoop?

- Scalable
- Cost effective
- Flexible
- Fast
- Resilient to failure

WORKFLOW



Analysis Using Hive

Data collected in this project gives us an idea of sentiments related to popular topics whether they are positive or negative. It also shows devices which have been used and other results like the user with maximum followers.

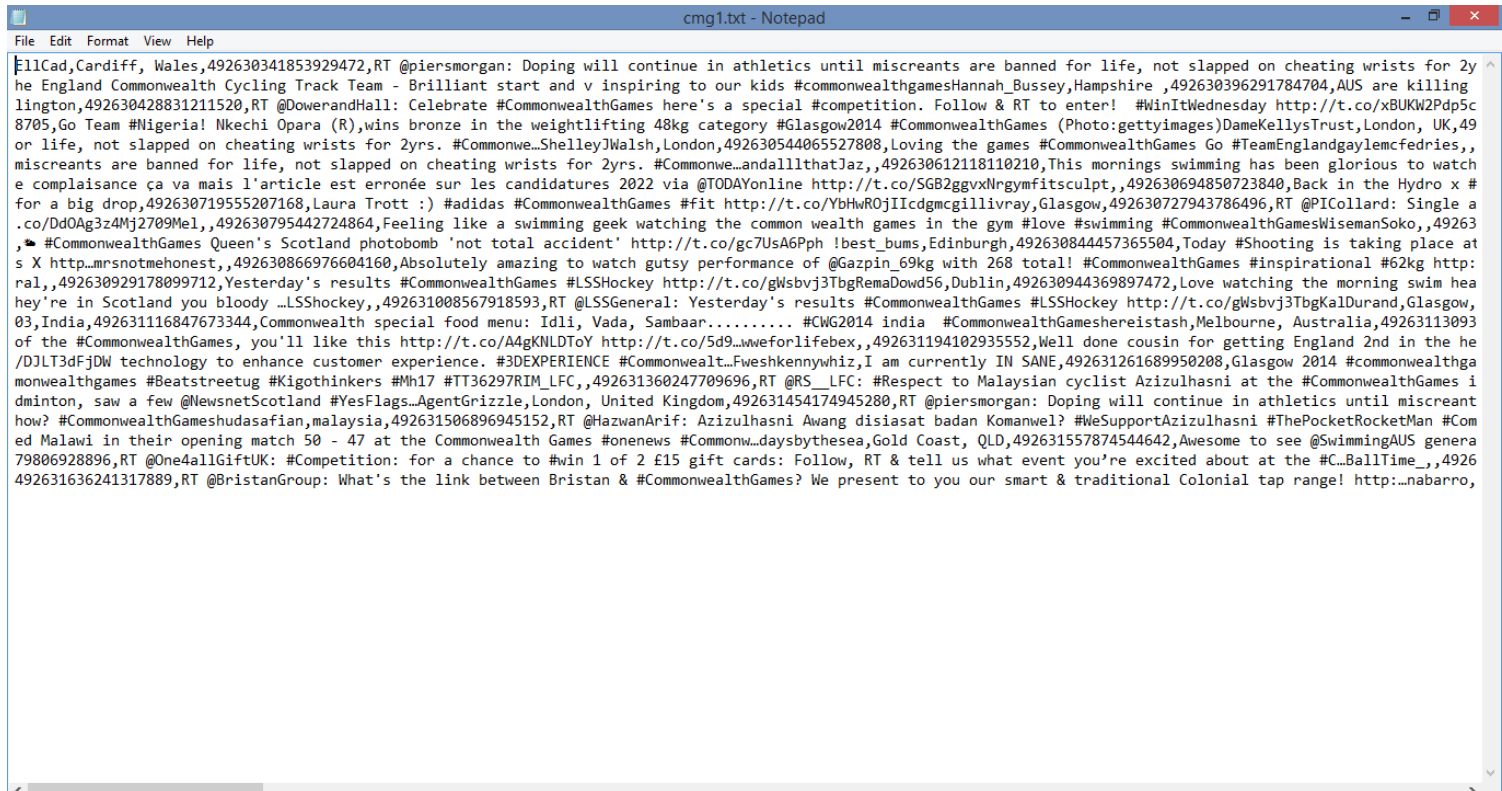
The most important aspect of this project is gathering all the data from twitter and preparing it for processing.

Data Gathering:

1. Data is aggregated from Twitter using java Twitter Streaming API.
2. To access Twitter Streaming API a developer account has to be created on twitter first. This gives a consumer key, consumer secret, access token and an access token secret. This enables authorised streaming of data. Open authentication (OAth) is used
3. Twitter4j java library is downloaded for importing functions
4. A configuration builder class is defined which authenticates data streaming.
5. Then to start the streaming of data the status listener class is defined which is called as soon as there is an incoming tweet.
6. All of the tweets are written into a text file named tweets.txt after being streamed. The writer class is defined for this purpose.
7. Various classes are defined which give the id of the user who tweeted, the device used, the timestamp of tweet, number of people who added tweet as favourite, number of followers the user has, number of people who retweeted it. This retweet is also streamed.
8. Aggregated raw data is cleansed using replace function to some extend before analysing it using Map-Reduce methods.
 - ☐ Removal of hyperlinks
 - ☐ Removal of punctuations
 - ☐ Removal of extra delimiters

9. The format of collected tweets has following fields:

- Id
- Created at
- Source
- Favourite Count
- Retweet Count
- Retweeted Text
- Username of person retweeting
- Screen name of the person retweeting
- Screen name of the user
- Tweet text
- No. of friends
- No. of followers
- No of statuses
- Verified (or not)
- Time zone
- Time zone offset(Coordinated Universal Time)



Raw Tweets Sample file (streamed From Twitter)

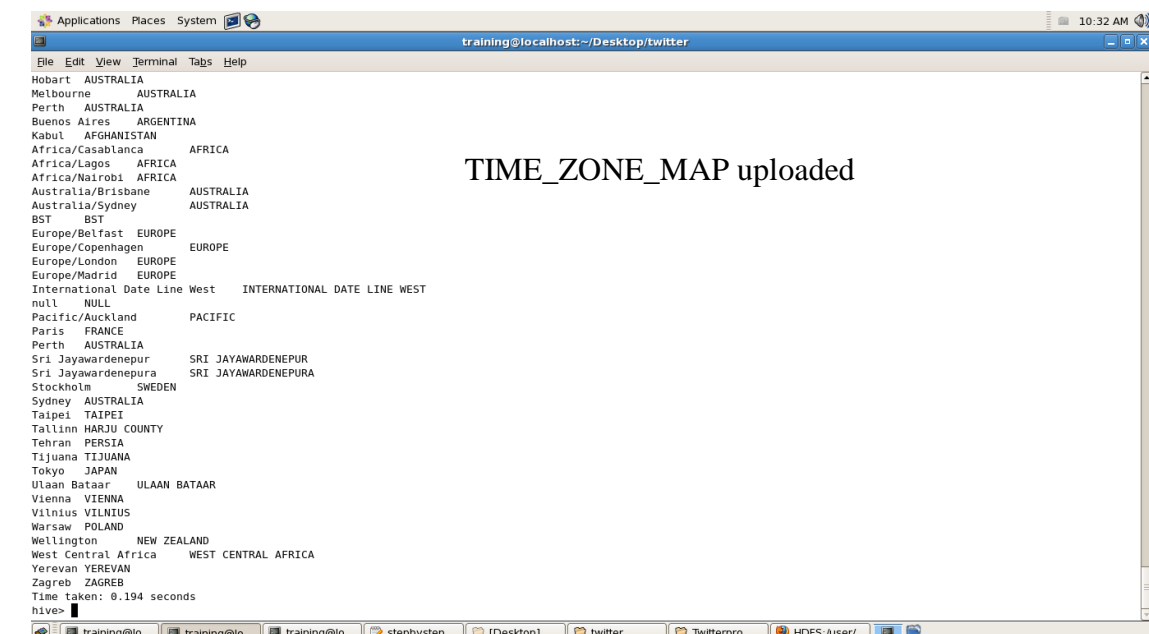
We first use the program to stream to run and test the code before actually implicating on larger data. Large data files cannot be opened in notepad. Directly using main files can cause wastage of time if there is any error on code.

DATA UPLOADING

We upload all data to the Hadoop Distributed File System(HDFS) as the hive component used by us to analyzes data supports only Hadoop Distributed File System.

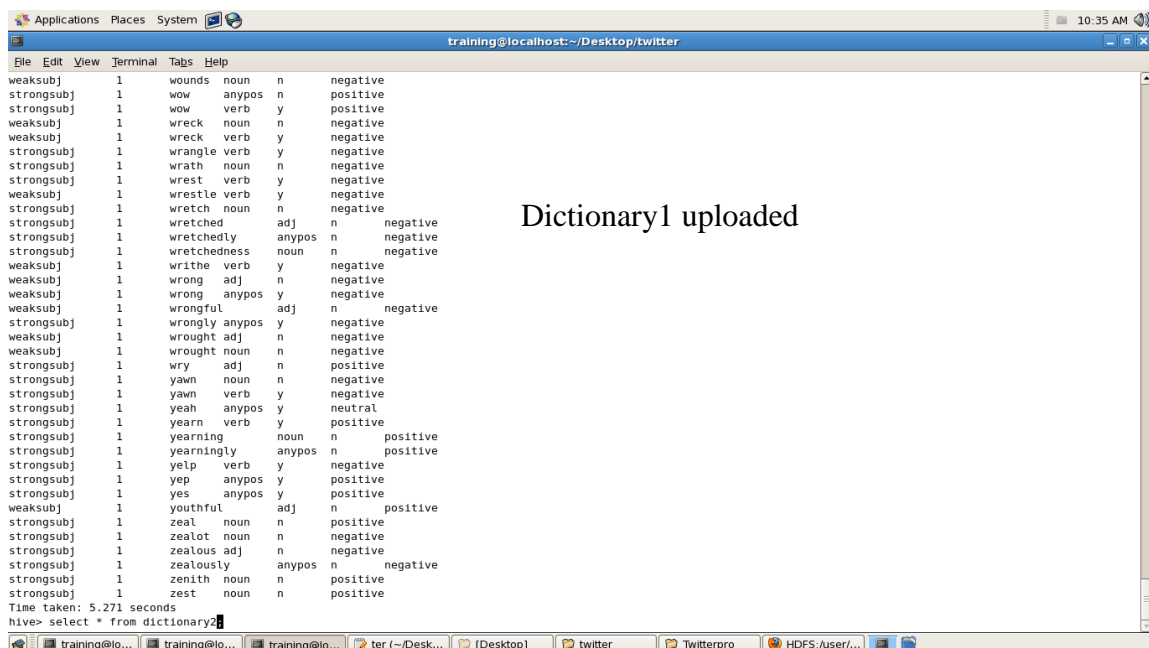
Data to HDFS:

1. Directories are created in HDFS each for tweets, dictionaries and timezone map.
2. By using the put command on Hadoop terminal we upload data
 - a. Tweets.txt
 - b. Dictionary1 and afinn-111 (dictionary)
 - c. Time zone mapinto their respective directories in HDFS.



A terminal window titled 'training@localhost:~/Desktop/twitter' shows the execution of the 'put' command to upload 'TIME_ZONE_MAP' to HDFS. The command is 'put /home/training/Desktop/timezone_map.txt /user/training/HDFS/'. The output shows a list of files being uploaded, including various location and time zone mappings like 'Hobart AUSTRALIA', 'Melbourne AUSTRALIA', 'Perth AUSTRALIA', 'Buenos Aires ARGENTINA', 'Kabul AFGHANISTAN', 'Africa/Casablanca AFRICA', 'Africa/Lagos AFRICA', 'Africa/Nairobi AFRICA', 'Australia/Brisbane AUSTRALIA', 'Australia/Sydney AUSTRALIA', 'BST BST', 'Europe/Belfast EUROPE', 'Europe/Copenhagen EUROPE', 'Europe/London EUROPE', 'Europe/Madrid EUROPE', 'International Date Line West INTERNATIONAL DATE LINE WEST', 'null NULL', 'Pacific/Auckland PACIFIC', 'Paris FRANCE', 'Perth AUSTRALIA', 'Sri Jayawardenepur SRI JAYAWARDENEPUR', 'Sri Jayawardenepura SRI JAYAWARDENEPURA', 'Stockholm SWEDEN', 'Sydney AUSTRALIA', 'Taipei TAIPEI', 'Tallinn HARJU COUNTY', 'Tehran PERSIA', 'Tijuana TIJUANA', 'Tokyo JAPAN', 'Ulaan Bataar ULAAN BATAAR', 'Vienna VIENNA', 'Vilnius VILNIUS', 'Warsaw POLAND', 'Wellington NEW ZEALAND', 'West Central Africa WEST CENTRAL AFRICA', 'Yerevan YEREVAN', and 'Zagreb ZAGREB'. The upload completes with the message 'Time taken: 0.194 seconds' and 'hive>'.

```
Applications Places System 10:32 AM
training@localhost:~/Desktop/twitter
File Edit View Terminal Tabs Help
Hobart AUSTRALIA
Melbourne AUSTRALIA
Perth AUSTRALIA
Buenos Aires ARGENTINA
Kabul AFGHANISTAN
Africa/Casablanca AFRICA
Africa/Lagos AFRICA
Africa/Nairobi AFRICA
Australia/Brisbane AUSTRALIA
Australia/Sydney AUSTRALIA
BST BST
Europe/Belfast EUROPE
Europe/Copenhagen EUROPE
Europe/London EUROPE
Europe/Madrid EUROPE
International Date Line West INTERNATIONAL DATE LINE WEST
null NULL
Pacific/Auckland PACIFIC
Paris FRANCE
Perth AUSTRALIA
Sri Jayawardenepur SRI JAYAWARDENEPUR
Sri Jayawardenepura SRI JAYAWARDENEPURA
Stockholm SWEDEN
Sydney AUSTRALIA
Taipei TAIPEI
Tallinn HARJU COUNTY
Tehran PERSIA
Tijuana TIJUANA
Tokyo JAPAN
Ulaan Bataar ULAAN BATAAR
Vienna VIENNA
Vilnius VILNIUS
Warsaw POLAND
Wellington NEW ZEALAND
West Central Africa WEST CENTRAL AFRICA
Yerevan YEREVAN
Zagreb ZAGREB
Time taken: 0.194 seconds
hive>
```



A terminal window titled 'training@localhost:~/Desktop/twitter' shows the execution of the 'put' command to upload 'Dictionary1' to HDFS. The command is 'put /home/training/Desktop/dictionary1.txt /user/training/HDFS/'. The output shows a list of words and their sentiment scores, such as 'weaksubj 1 wounds noun n negative', 'strongsubj 1 wow anypos n positive', 'strongsubj 1 wow verb y positive', 'weaksubj 1 wreck noun n negative', 'weaksubj 1 wreck verb y negative', 'strongsubj 1 wrangle verb y negative', 'strongsubj 1 wrath noun n negative', 'strongsubj 1 wrest verb y negative', 'weaksubj 1 wrestle verb y negative', 'strongsubj 1 wretch noun n negative', 'strongsubj 1 wretched adj n negative', 'strongsubj 1 wretchedly anypos n negative', 'strongsubj 1 wretchedness noun n negative', 'weaksubj 1 writhe verb y negative', 'weaksubj 1 wrong adj n negative', 'weaksubj 1 wrong anypos y negative', 'weaksubj 1 wrongful adj n negative', 'strongsubj 1 wrongly anypos y negative', 'weaksubj 1 wrought adj n negative', 'weaksubj 1 wrought noun n negative', 'strongsubj 1 wry adj n positive', 'strongsubj 1 yawn noun n negative', 'strongsubj 1 yawn verb y negative', 'strongsubj 1 yeah anypos y neutral', 'strongsubj 1 yearn verb y positive', 'strongsubj 1 yearning noun n positive', 'strongsubj 1 yearningly anypos n positive', 'strongsubj 1 yelp verb y negative', 'strongsubj 1 yep anypos y positive', 'strongsubj 1 yes anypos y positive', 'weaksubj 1 youthful adj n positive', 'strongsubj 1 zeal noun n positive', 'strongsubj 1 zealot noun n negative', 'strongsubj 1 zealous adj n negative', 'strongsubj 1 zealously anypos n negative', 'strongsubj 1 zenith noun n positive', and 'strongsubj 1 zest noun n positive'. The upload completes with the message 'Time taken: 5.271 seconds' and 'hive> select * from dictionary2;'.

```
Applications Places System 10:35 AM
training@localhost:~/Desktop/twitter
File Edit View Terminal Tabs Help
weaksubj 1 wounds noun n negative
strongsubj 1 wow anypos n positive
strongsubj 1 wow verb y positive
weaksubj 1 wreck noun n negative
weaksubj 1 wreck verb y negative
strongsubj 1 wrangle verb y negative
strongsubj 1 wrath noun n negative
strongsubj 1 wrest verb y negative
weaksubj 1 wrestle verb y negative
strongsubj 1 wretch noun n negative
strongsubj 1 wretched adj n negative
strongsubj 1 wretchedly anypos n negative
strongsubj 1 wretchedness noun n negative
weaksubj 1 writhe verb y negative
weaksubj 1 wrong adj n negative
weaksubj 1 wrong anypos y negative
weaksubj 1 wrongful adj n negative
strongsubj 1 wrongly anypos y negative
weaksubj 1 wrought adj n negative
weaksubj 1 wrought noun n negative
strongsubj 1 wry adj n positive
strongsubj 1 yawn noun n negative
strongsubj 1 yawn verb y negative
strongsubj 1 yeah anypos y neutral
strongsubj 1 yearn verb y positive
strongsubj 1 yearning noun n positive
strongsubj 1 yearningly anypos n positive
strongsubj 1 yelp verb y negative
strongsubj 1 yep anypos y positive
strongsubj 1 yes anypos y positive
weaksubj 1 youthful adj n positive
strongsubj 1 zeal noun n positive
strongsubj 1 zealot noun n negative
strongsubj 1 zealous adj n negative
strongsubj 1 zealously anypos n negative
strongsubj 1 zenith noun n positive
strongsubj 1 zest noun n positive
Time taken: 5.271 seconds
hive> select * from dictionary2;
```


DATA PROCESSING IN HIVE

1. Now all the required data is uploaded into hdfs so now we create Tables in hive.
2. First we create a table raw_tweets and load all the raw tweets gathered from Twitter In the following format:

```
CREATE TABLE raw_tweets (  
  id BIGINT,created_at STRING,  
  source STRING,favorited int,retweet_count INT,  
  Retweeted_text STRING, retweeted_username STRING,  
  Retweeted_userscreenname STRING,  
  screen_name STRING,text STRING,  
  name STRING,friends_count INT,  
  followers_count INT,statuses_count INT,  
  verified BOOLEAN,utc_offset INT,  
  time_zone STRING  
)  
ROW FORMAT delimited fields terminated by ',';
```

3. Then table Dictionary1, Dictionary 2, Time Zone Map and load respective file stored in hdfs into the tables.
4. Now since the raw_tweets table has lot of data we start to create different views to get Different parts of data for various types of analysis.
5. We start by doing the Sentiment analysis on the tweets. For this we require only the text of the tweet, time zone and its id.
6. The ID field acts as a Key when forming key value pairs in the mapreduce program and is used for grouping the tweets and identifying which word originated from which tweet at later states.
7. So for this a view is created which selects only the tweets' text id and time zone from raw tweets named as plain tweets.

Example:

Id	Text	Time Zone
424653423245	This is a Twee	New Delhi

8. Another View is created called tweet_country is created it is used to find the the country from the time zone. For this the time zone map table is joined from left side With the plain tweets table with the condition that id is same in both the tables.

Example:

Id	Text	Time Zone	Country
424653423245	This is a Twee	New Delhi	India

9. Now a view tweets array is created which shows the words of the tweets of an id as a single array of words. It basically breaks down the sentence into words and creates an array. The words are all converted to lower case then passed to the explode function. Then we create a lateral view of all word of an id.

Id	Text
424653423245	["this","is","a","tweet"]

10. The next view created tweet_row further breaks down the array. Each word of a user is written in a single row along with the id acting as key

Id	Text
424653423245	this
424653423245	is
424653423245	a
424653423245	tweet

11. Now the tables of dictionaries are used to compute the sentiment of each word by using the left side join operation of hive and sentiment of each word is added after it.

Id	Text	Score
424653423245	this	+1
424653423245	is	-1
424653423245	a	-2
424653423245	tweet	-3

(the scores used above are for example not actual scores)

The above table is called computed

12. By using left outer join on tweet_row and dictionary1 every word in the table a view computed_1 is created. It has a score associated to it by matching the same words on dictionary1. If the word is not present in dictionary1 it is marked as 0.
13. For dictionary2 already a score is present for each word. By using left outer join, on tweet_row and dictionary2, a view computed_2 is created.
14. Now we combine the rows in the computed table and group them by id and sum the tweets to find the sentiments.
15. Tables sentiment1 and sentiment2 are created respectively from computed_1 and computed_2. The score for each id is added using sum function and group by id function for computed_2 and saved in the first table. Whereas for computed_1 the polarity for each id is added and sentiment is added as positive for score greater than zero and vice versa (for zero sentiment is zero).
16. The result table also has a column for country which is mapped using left outer join on time zone map. Also for sentiment _result1 (for dictionary 1) the sentiment 'positive' is replaced by score +1 and so on.

17. In the end the column for ids is removed to give the final table grouped by country by adding all scores of a particular country.
18. Also we create a view user profiles which contains information of the user like his screen name, friends count, followers count, status count and whether it is verified or not.

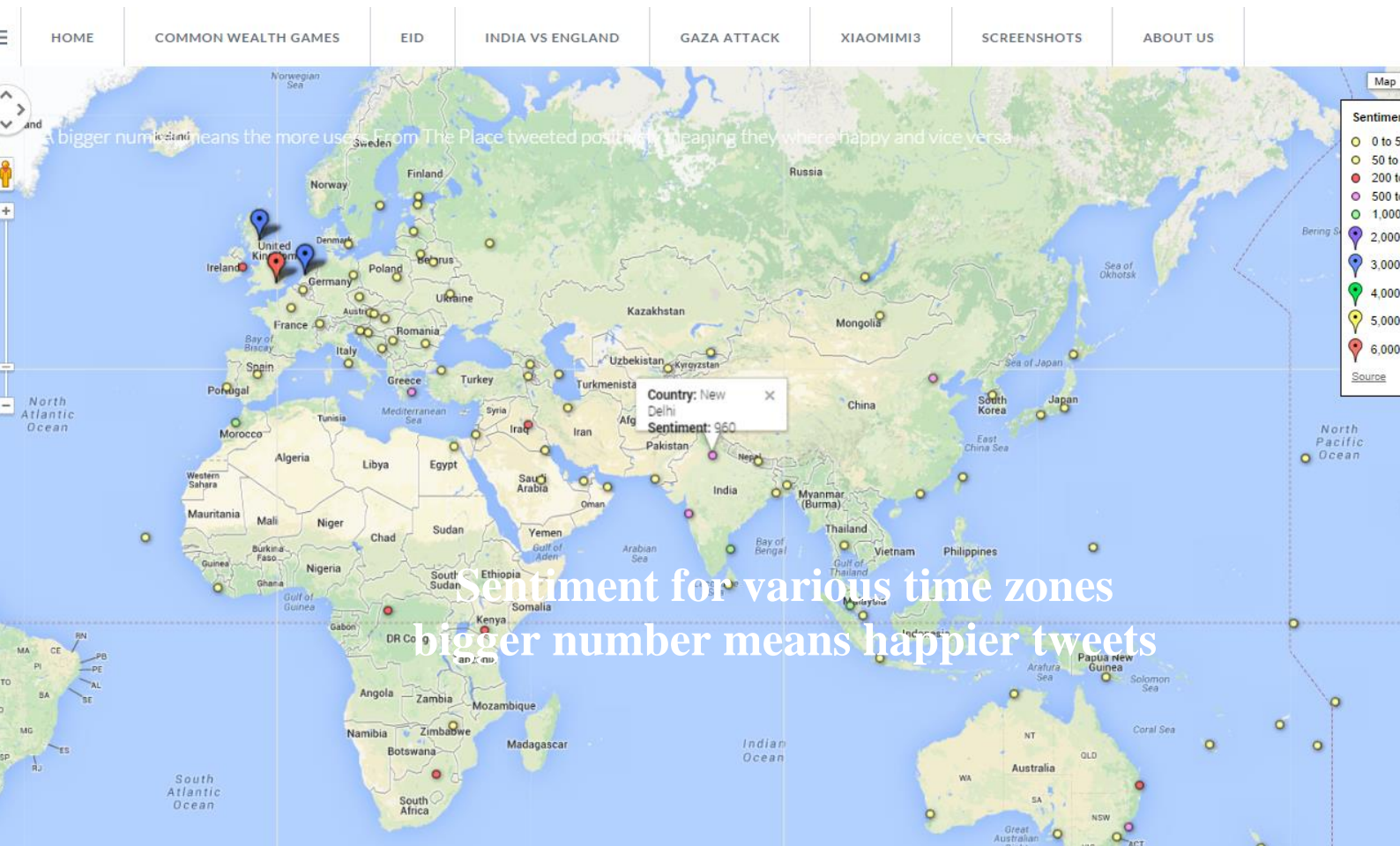
The above steps are repeated for each of the topic being analyzed.

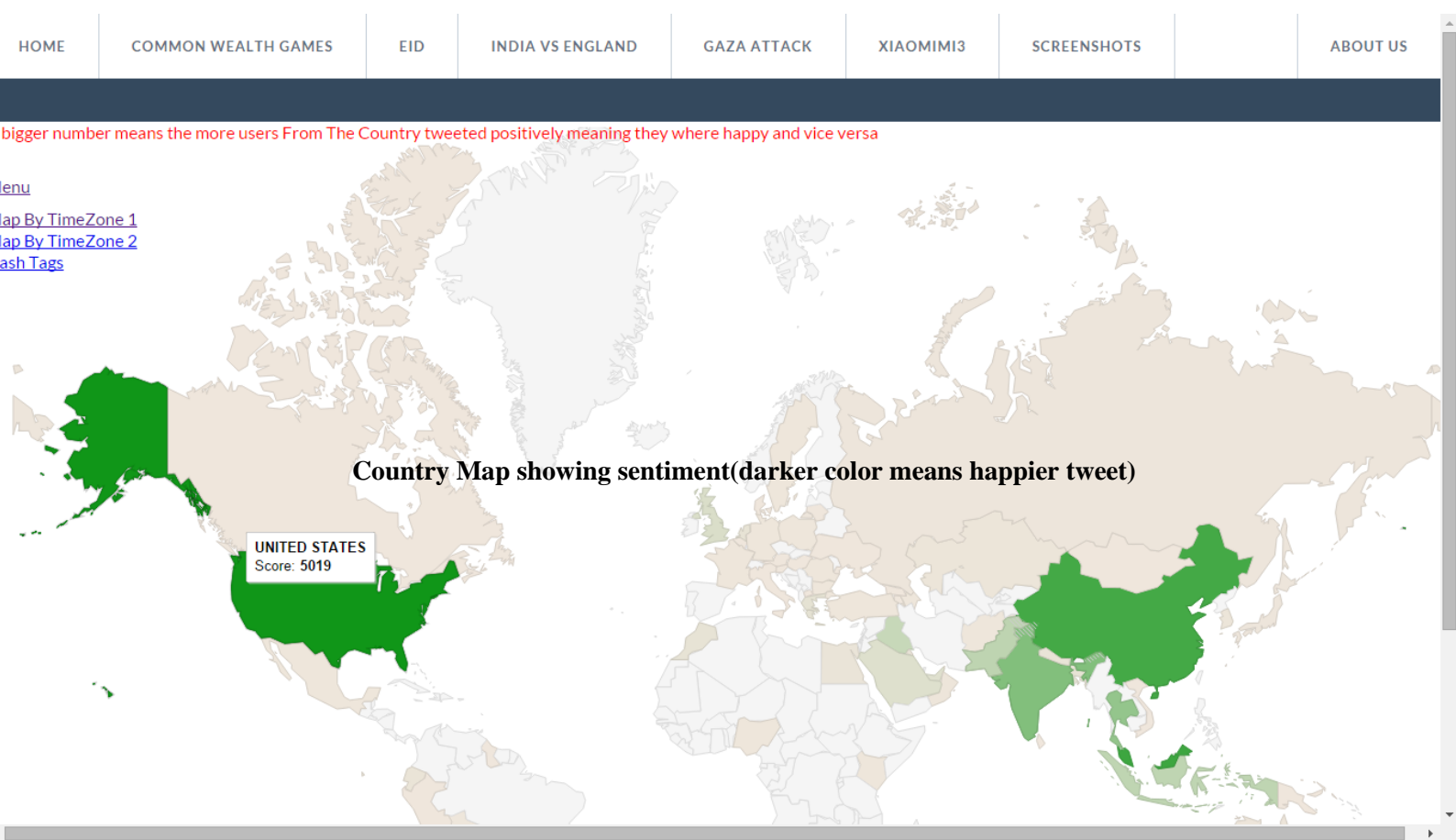
DATA VISUALIZATION

1. The output table for the required data are extracted from hive warehouse and stored in HDFS.
2. The output extracted include: Sentiment tables, user profile table, time zone tables.
3. The extracted output is then copied from hdf5 to local machine.
4. The output data is the visualized using various tools.
 - a. Google Maps API
 - b. Google Visualisation API
 - c. HTML/CSS tags.

DATA Presentation

The Visualised data is put together and represented in the form of a website Twitter data Analysis.







commonwealth games

#bringiton

#sayateammalaysia

#cwg2014

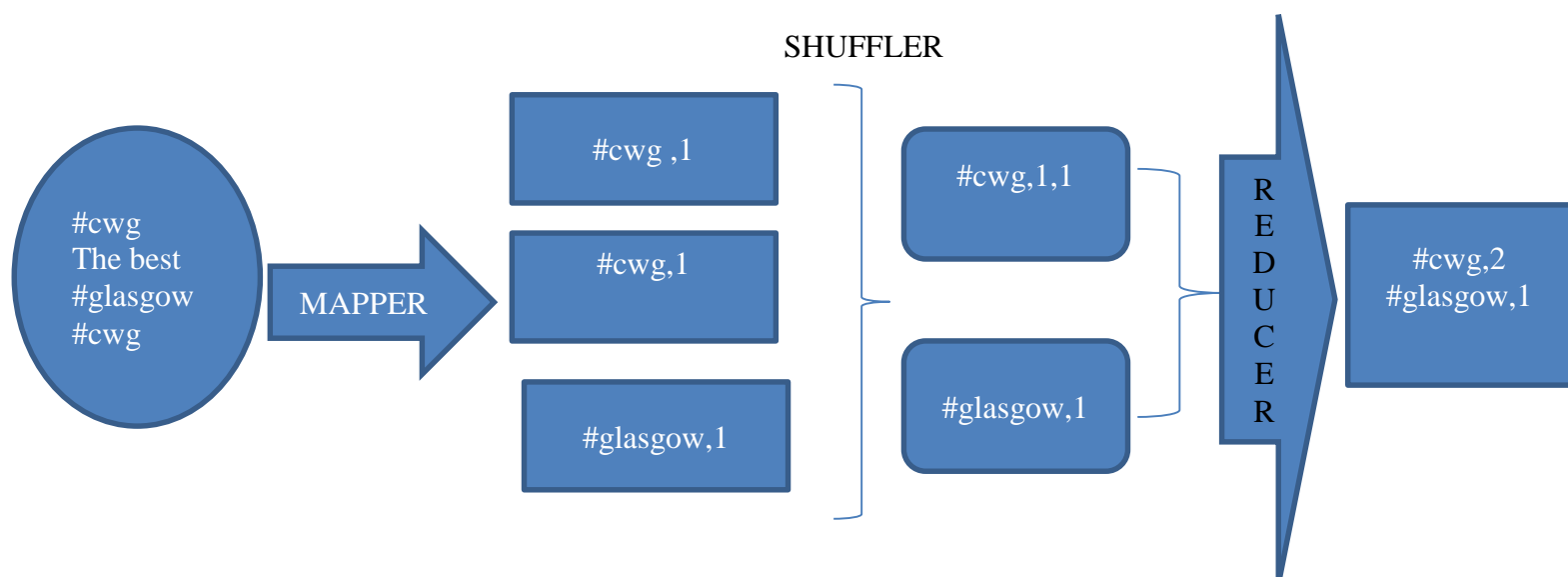
#diving
#glasgow
#teamscotland
#sukant
#badminton
#teamengland

Project by

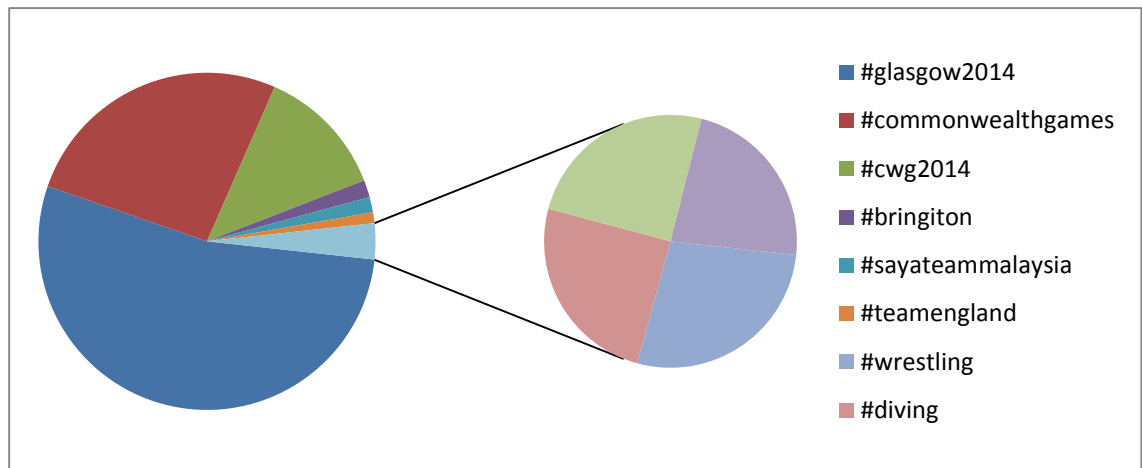
Nitesh Sarah Rahul

ANALYSIS HASHTAGS USING MAP REDUCE PROGRAM

1. Data is collected in the same manner as we had done for analysis using hive.
2. The map reduce job can only run on a file saved in HDFS
3. The streamed data is saved in the text file and uploaded to HDFS.
4. The map reduce program counts the words beginning with hash tags.
5. The mapper program identifies a word beginning with '#' converts the word to lower case and passes it to the reducer program.
6. Example:



7. This mapper class output is passed to shuffler first which groups the words according to its count.
8. Finally the mapper class gives the output which contains two fields- the word and its no. of occurrences.
9. The analysis of this word count can be presented as a pie chart:



LIMITATIONS

- The limitation of this project can be that the data which is processed is not live. The data being streamed and stored in the text file is live. But this gathered/collected data is further processed to compute results hence it is not live.
- Regarding sentiment analysis it isn't complete yet. It is still in its infancy, and there are limitations. Despite significant advances in machine learning, it's extremely difficult (or not practically efficient) for computers to understand and process natural language, automate sentiment analysis, or determine ambiguous context.
- Twitter has users from all over the world but all the tweets are not being processed. This is because foreign scripts are not detected. The dictionaries used have only English words with their sentiment scores. Tweets with other scripts are filtered out.
- The problem is that the more you break down the data, the less likely it is that automated analysis will get it right. Automated sentiment works best with large amounts of data and can't be relied upon for smaller samples. A lot of humans struggle with sarcasm and irony, so how can we expect computers to cope? This is particularly problematic when looking at Twitter. There are also examples of posts which contain both positive and negative sentiment. For instance: "The food was fantastic but the service was terrible". In this case a computer won't know which way to turn.

CONCLUSION

This project gave us hands on experience of handling and parallel processing of huge amount of data. Data collection process introduced us to java twitter streaming API. It was very interesting to gather and then aggregate the social networking data so as to extract interesting patterns and recent trends from it. We got exposure to work with prominent parallel data processing tool: Hadoop. Apache Hadoop framework is gaining significant momentum from both industry and academia as the volume of data to analyze growth rapidly. This project helped us not only to gain knowledge about installation and configuration of hadoop distributed file system but also map reduce programming model. At the end of analysis phase data visualization was performed with the help of Google Developer.

Amongst the many fields of analysis, there is one field where humans have dominated the machines more than any – the ability to analyze sentiment, or sentiment analysis. The future of this data analysis field is vast. IBM has announced the launch of its new API Watson which can give 86% accuracy of the sentiments.

This project not only analyses the sentiments of the user but also computes other results like the user with maximum friends/followers, top tweets etc. hence hadoop can also be effectively used to compute such results in order to determine the current trends with respect to particular topics. This can be very useful in the marketing sector. But sentiment analysis is bigger than Twitter, Facebook, and the rest of social media's domain, as myriad professionals try to analyze massive amounts of internal customer data too.

Bibliography and References

1. Twitter API
2. Cloudera.com
3. HortonWorks.com
4. Google Fusion table
5. Google Visualization API
6. Finn Arup Nielsen , <http://arxiv.org/abs/1103.2903>
7. <http://Twitter4j.org>
8. Apache Hadoop
9. <http://tympanus.com/codrops>