

multirow



NIMRAH BUTT
01-134131-072

Social Media Analytics

Bachelor of Science in Computer Science

Supervisor: Dr. Muhammad Muzammal

Department of Computer Science
Bahria University, Islamabad

November 2016

Certificate

We accept the work contained in the report titled “Social Media Analytics”, written by Ms. Nimrah Butt as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by . . . :

Supervisor: Dr. Muhammad Muzammal (Associate Professor)

Internal Examiner: Name of the Internal Examiner (Title)

External Examiner: Name of the External Examiner (Title)

Project Coordinator: Dr. Arif ur Rahman (Assistant Professor)

Head of the Department: Dr. Faisal Bashir (Head of Department)

December 6st, 2016

Abstract

This project highlights the importance of growing data and the need to use it effectively. Social media gives everyone a good and a remarkable open door for connecting to the outside world. Social Media Analytics provides us a tool through which we gather and download data from different social sites and perform sentiment analysis on the respective data. The purpose of this project is to analyze the new and latest tools and technologies which are rapidly growing and are being used in the world and how can we achieve our task using those technologies and tool. The goal is to approach social media and collect data which is growing and trending on social media and perform sentiment analysis on the data and showing the gather result under the hat of positive, negative or neutral. Hence we can achieve the purpose of having desired data and knowing the reaction of the outside world on it. The angles investigated in this project are the aptitudes and strategies required to accomplish the coveted objective keeping in mind the privacy of social media user.

Acknowledgments

I want to express my most profound gratefulness to my professor Dr. Muhammad Muza-mmah whose help, animating recommendations, support, encouragement, criticism and consolation gave me the strength and likelihood to finish this project. I might want to thank all the brilliant individuals whose support made this exploration a possibility. A special thanks and recognition to our final year project coordinator, Dr. Arif ur Rehman for making sure and supporting us to achieve our goals on respective time. I also offer my earnest gratefulness to Bahria University for the learning openings given by them who keep forming every understudy to go past their breaking points. My culmination of this project couldn't have been done without the support of my companions and company Metis, they are critical and trusted people who have upheld me all through my work, much appreciated! In conclusion I might want to thank my parents, who have been there for me all through this, all their exertion and trust on me has helped me achieve this milestone of my life.

NIMRAH BUTT
Islamabad, Pakistan

November 2016

*“We think someone else, someone smarter than us,
someone more capable, someone with more resources will solve that problem.
But there isn’t anyone else.”*

Regina Dugan

Contents

Abstract	i
1 Introduction	1
1.1 Project Overview	1
1.1.1 Social Media	2
1.1.2 Big Data	2
1.1.3 Sentiment Analysis	3
1.2 Problem Description	3
1.2.1 Academic Objectives	4
1.3 Product Objective	4
1.4 Project Objective	4
1.5 Project Scope	4
2 Literature Review	6
2.1 Overview	6
2.1.1 Big Data Technologies	6
2.1.2 HADOOP	7
2.1.3 HDFS	8
3 Requirement Specifications	10
3.1 Application Overview	10
3.2 Application System Environment	10
3.3 Basic Functionality	11
3.3.1 Twitter	11
3.3.2 Facebook	11
3.4 User Characteristics	12
3.5 Functional Requirements	12
3.6 Non-Functional Requirements	13
3.7 Hardware Selection	14
3.8 Performance Requirements	14
3.9 User Interface	14
3.9.1 Tools	15
3.10 Use Cases	15
4 Design	19
4.1 System Architecture	19
4.1.1 Twitter	19

4.1.2	Facebook	19
4.1.3	Sentiment Analysis	19
4.1.4	Dashboards	19
4.2	Deployment Diagram	20
4.3	System Sequence Diagram	21
4.4	High-level Diagram	22
4.5	Process Interaction Models	22
5	System Implementation	25
5.1	Tools and Technologies	25
5.1.1	Virtual Machine	25
5.1.2	Big Data	25
5.1.3	HADOOP	26
5.1.4	HADOOP Distributed File System <i>HDFS</i>	26
5.1.5	HIVE	26
5.1.6	Flume	27
5.1.7	Spark	27
5.1.8	Big Data Discovery	27
5.1.9	Oracle Data Integrator <i>ODI</i>	27
5.2	Languages	27
5.3	Methodology	27
5.3.1	Facebook	28
5.3.2	Twitter	40
5.3.3	Sentiment Analysis	43
5.3.4	Big Data Discovery	46
6	System Testing and Evaluation	49
6.1	Software Testing Techniques	49
6.1.1	Unit Testing	49
6.1.2	Integration Testing	50
6.1.3	System Testing	50
6.1.4	Structural Testing	50
6.1.5	Performance Testing	50
6.1.6	Stress Testing	50
6.1.7	Configuration Testing	50
6.1.8	Security Testing	51
6.1.9	Acceptance Testing	51
6.2	Test Cases	51
6.2.1	Test Case # 1	51
6.2.2	Test Case # 2	51
6.2.3	Test Case # 3	52
6.2.4	Test Case # 4	52
7	Conclusions	54
7.1	Limitations and Future Enhancement	54
A	User Manual	55

CONTENTS

vi

References

56

List of Figures

1.1	4 V's of Big Data	3
2.1	HADOOP	8
3.1	System Environment	11
3.2	Twitter Functionality	11
3.3	Facebook Functionality	12
3.4	User and System Interaction	15
3.5	Opening the Webpage	16
3.6	Twitter Access	16
3.7	Facebook Access	17
3.8	Sentiment Analysis	18
4.1	Describes the complete application architecture	20
4.2	System Deployment Diagram	20
4.3	System Sequence Diagram	21
4.4	High-level Diagram	22
4.5	Twitter Process Model	22
4.6	Facebook Process Model	23
4.7	Sentiment Analysis Process Model	24
5.1	4V's of Big Data	26
5.2	Facebook Developer Site	29
5.3	Facebook App: Project	29
5.4	Facebook App: Application ID	30
5.5	Facebook App: Product Setup	30
5.6	Facebook App: Token Generation	31
5.7	Facebook App: Application Security	31
5.8	Running the Virtual Machine	32
5.9	Oracle Big Data Lite	32
5.10	Oracle Big Data Lite: Local Domain	33
5.11	Oracle Big Data Lite: Content of Directory	33
5.12	Oracle Big Data Lite: Flume	34
5.13	Directories	34
5.14	Flume Agent	35
5.15	Sources and Sinks	36
5.16	Facebook Group Configuration	36
5.17	Facebook Shell Script	37

5.18 Facebook Group Test	38
5.19 Oracle VM	39
5.20 Flume Downloaded Files	39
5.21 Twitter Developer Documentation	40
5.22 Twitter Developer Documentation	41
5.23 Flume Configuration	42
5.24 Twitter Agent	43
5.25 Sentiment Analysis	46
5.26 Big Data Discovery	46
5.27 Oracle Data Integrator	47
5.28 Generating Scenarios	47
5.29 Facebook Scenario	48

List of Tables

3.1	Use Case 1	16
3.2	Use Case 2	17
3.3	Use Case 3	17
3.4	Use Case 4	18
4.1	Process Model 1	23
4.2	Process Model 2	23
4.3	Process Model 3	24
6.1	Test Case 01	51
6.2	Test Case 02	52
6.3	Test Case 03	52
6.4	Test Case 03	53

Acronyms and Abbreviations

API	Application Programming Interface
DStream	Discretized Stream
HQL	Hive Query Language
HDFS	Hadoop Distributive File System
ROI	Return on Investment
RDD	Resilient Distributed Dataset
ODI	Oracle Data Integrator

Chapter 1

Introduction

1.1 Project Overview

In this 21st century as arrival of new technologies every day is eminent, software, devices and all the other communication means which includes social networking sites, the amount of data produced by human being is growing very rapidly. Most of the information is now available on internet social networking sites, resulting in increase in data on daily bases. It is evidently said that now we are creating 2.5 quintillion bytes' data every day. Talking about the rapid change in technology it is said that in today's world 90% of the data over the internet is solely from the last two years alone and this rate is still growing enormously. The whole world is revolving around data. As we know every field has some sort of data. Data streams are everywhere form smaller level to bigger level. The expanding volume and differing qualities of information make it to a great degree testing to store, recover, examine and use this data when required. Organizations, government offices, and society require the learning to outline, create and convey complex data frameworks and applications that arrangement with multi-terabyte information sets. The new technologies are offering us more than what we can imagine. These associations are characterizing new activities and reexamining existing systems to inspect how they can change their organizations utilizing Big Data. We can play with the data and extract useful information from it. Nowadays social media plays a very impactful and inspirational role in our daily life. Now there are some organizations which needs to get some track of all kind of information about an individual for their own safety or a basic analysis of tweets, re-tweets, likes, comments, post to develop an in-depth idea of the social consumer, see the trends and can generate an idea of what kind of data is spinning around the social media and what is the impact of that data and what information can we extract from it.

1.1.1 Social Media

A social media is an online platform that is provided to people to build their networks, connection, share content and collaboration. It provides you to be online connected and interact with people all over the world regardless of geographical distance. Social media is gaining the traffic and attention of people all around the globe using some social media sites. Every site is different from one another. Every site is providing thoroughly diverse social activities. For instance, Twitter is a social media site intended in a way that people can exchange minimal posts with others. On the other hand, Facebook, is a full-scale social networking site that permits users to perform variety of activities which includes sharing updates, photos, joining events etc. Every social site has its own norms and conditions which is somehow different from each other in some aspect. Inside the social media the main thing is DATA. Large amount of data which is growing tremendously on the daily bases.

1.1.2 Big Data

In many researches it is clearly mentioned that the term big data was coined to capture the meaning of this emerging trend of growing data and refers to technologies and initiatives that involve data that is too diverse, fast-changing or massive for conventional technologies, skills and infra- structure to address efficiently, which includes all social media websites. Social media is growing as an area of information technology and service. Social media contains all kind of data, large data which comes under the term “Big data”. Keeping in mind today’s world the concept of social sites is growing and with that Big data is also growing and its impact in near future will be more important as Big Data is today, its impact will be even more notable. The reason behind this is the Big data itself. It defines the whole generic set of data.

- 4V’s of Big Data

Following are the four V’s of Big Data:

- **Volume:** It implies huge measure of information, essentially associations gather information from an assortment of sources, including business exchanges, online networking and data from sensor or machine-to-machine information. Previously, putting away this expansive measure of information was a major issue however because of this developing eld putting away huge measure of information is not an issue. (E.g. HADOOP will discuss this later).
- **Velocity:** It implies information touches base at fast, data streams in at a mysterious speed and should be managed in a convenient way as time is one of the fundamental imperative.

- **Variety:** It implies information originates from blended assets. For example, information comes in a wide range of configurations, for example, organized, instructed, numeric information, databases, unstructured content archives, email, video, sound, stock ticker information and budgetary exchanges. All these are interlinked with social websites directly. Social sites work with big data.
- **Veracity:** It implies the accuracy of the data. Certainty of the data we are using. Veracity refers to collected data and to which degree the data can be trusted [1].

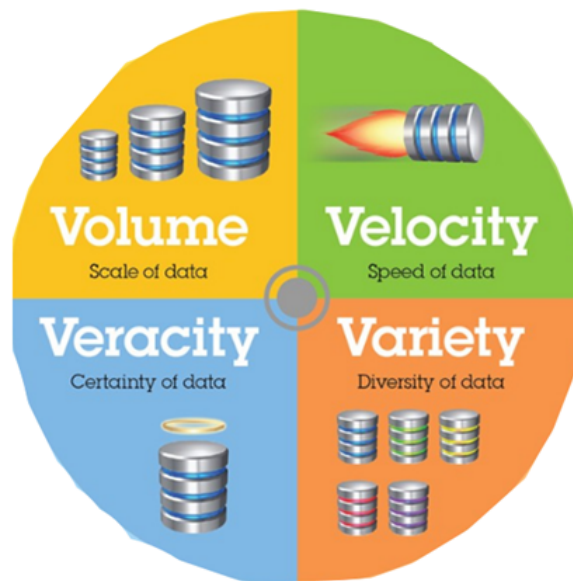


Figure 1.1: 4 V's of Big Data

All these are interlinked with social websites directly. Social sites work with big data. What can be done using that data?

1.1.3 Sentiment Analysis

Sentiment analysis is also known as opinion mining. It is the process of determining the emotional tone behind the written words. Whether the written words are positive, negative or neutral. We use sentiment analysis so that we can understand the attitude, meaning, opinions, likes, dislikes, emotions, hatred, love, sarcasm positivity, negativity behind the words people have written. Sentiment analysis used in social media is of a vital use because it gives us wider public opinion regarding certain topic.

1.2 Problem Description

As the world and technology is growing so rapidly there was an urge for organization to gather data for multiple purposes. Company need to know their market audience, users

and competitors. The company adopts a practice we often called sentiment analysis to have the ability to extract insights from a social media and to monitor their business via social media. Social media allows company to keep their fingers on the pulse of what's important to their customers to grow in their interests. The company needs to use analytics to monitor their business via social media, by which they can also achieve a better return on investment *ROI*. Basically it is a percentage and is typically used for personal financial decisions, to compare a company's profitability or to compare the efficiency of different investments. Analytics will help company to allocate scarce resources, optimize market outcomes or limit risks.

1.2.1 Academic Objectives

The academic objective of this project is to learn new technologies, tools and a new and latest programming language. Research about the new emerging terms which will take over the world soon enough. To know the inside of how the new technologies can help us in growing and make things more advance and available on one click.

1.3 Product Objective

The objective of this project is to extract data from social sites i.e. Twitter and Facebook and perform sentiment analysis on the data and showing whether the data is positive, negative or neutral and what results we can extract from the final information. When we say social media the term Big data is automatically emerged because the data we are talking about is massive. We use new ongoing framework HADOOP for purpose of extracting data from social sites.

1.4 Project Objective

The objective of this project is to deliver a complete insight of what is going on the social media, what are the trends and growing interest of people, what one can do to make better decisions about the company and the steps he is going to take in future, where he can invest according to the need of the world and new technologies. As this is an industrial project this will help the company to grow in the right direction keeping their audience, people, latest trends, feedback, praises, negativity all kind of feedback in mind.

1.5 Project Scope

Social media analytics system will allow organization to extract data from social sites convert it into useable data in i.e. the information in real time and then performing

sentiment analysis on that data and generating the dashboards showing the results. At first the scope of this project was to extract data only from Twitter and perform sentiment analysis on it, but now we will also extract data from Facebook. In Facebook for now we will extract data from a specific Group and page only. Solving big data analytics challenges requires a complete ecosystem. There are some constraints involved which are further linked with risk involved in our system. The software tool we are using for extracting the data from social sites is HADOOP which is an open source framework. Its installation is very expensive. No tools other than Oracle will support our system. The minimum RAM required for the tools to run is 12.0 GB not every processor supports this e.g. Intel Core i3 does not support 12.0 GB due to which the tool won't run. The Virtual Machine and all the tools are using maximum RAM this PC supports hence the process takes time when run on local net otherwise when the process is run on server with faster coverage of net the time taken is minimized. The sentiment analysis and dashboard generation will take 9 to 10 minutes to complete. As the memory of this machine is at its maximum sometimes dashboards also need to be refreshed at least 2 to 3 times to display meanwhile on large server it takes only the click. For now, our project will extract data from Twitter via HASHTAGS and data from Facebook's specific group and page. People's profile on Facebook and any other social site will be out of scope for this project.

Chapter 2

Literature Review

2.1 Overview

As we are using new technologies and tool in this project we need to explore every aspects of technologies. We will be extracting data from social sites using HADOOP. We will further be doing sentiment analysis on the data we have extracted using spark streaming and then generating the dashboards using Big Data Discovery. There are a lot of big data technologies. We will be working on some of it extracting knowledge from the already existing technologies.

2.1.1 Big Data Technologies

Huge information advancements are fundamental in giving more correct examination, which may incite more strong essential authority realizing more unmistakable operational efficiencies, cost diminishments, and decreased threats for the business. To harness the drive of enormous information, you would require a base that can direct and get ready titanic volumes of sorted out and unstructured information progressively and can guarantee information assurance and security. There are distinctive advancements in the market from different dealers including Amazon, IBM, Microsoft, et cetera to handle enormous information. While examining the advancements that handle huge information, we inspect the accompanying two classes of innovation:

- **Operational Big Data:** This incorporate systems like MongoDB that give operational capacities to ongoing, intuitive workloads where data is essentially caught and put away. NoSQL Big Data systems are intended to exploit new distributed computing models that have developed over the previous decade to permit huge calculations to be run cheaply and proficiently. This makes operational big data workloads much less demanding to oversee, less expensive, and quicker to actualize. Some NoSQL

systems can give bits of knowledge into examples and patterns in view of ongoing data with negligible coding and without the requirement for data researchers and extra framework.

2.1.2 HADOOP

We are using Apache HADOOP which is an open source software for all kind of computing. It gives monstrous capacity to any sort of information, gigantic data processing power and the capacity to handle for all intents and purposes boundless simultaneous undertakings or tasks.

1. Storage and processing speed It has Capacity to store and process immense measures of any sort of data, rapidly. With information volumes and assortments continually expanding, particularly from social networks.
2. Powerful Hadoop's conveyed figuring model procedures huge information quick. The all the more figuring hubs you utilize, the all the more preparing force you have.
3. Low cost It is open-source framework
4. Fault tolerance Information and application handling are ensured against equipment disappointment. In the event that a hub goes down, occupations are naturally diverted to different hubs to ensure the appropriated figuring does not fall flat.
5. Flexibility Dissimilar to conventional social databases, you don't need to preprocess information before putting away it. You can store as much information as you need and choose how to utilize it later. That incorporates unstructured data as well.
6. Scalability You can without much of a stretch develop your framework to handle more information just by including hubs. Little organization is required [2].

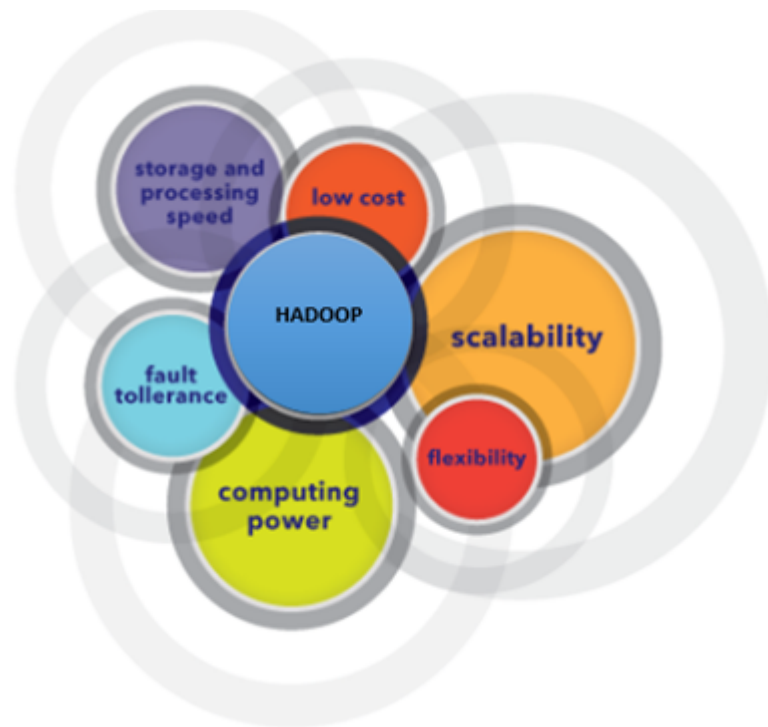


Figure 2.1: HADOOP

2.1.3 HDFS

The Hadoop Distributed File System (HDFS) is a sub-undertaking of the Apache Hadoop venture. This Apache Software Foundation task is intended to give a shortcoming tolerant document framework intended to keep running on product equipment. It is used to store the large amount of data and to stream those data or information sets at high transfer speed to client applications. HDFS stores filesystem metadata and application information independently. The HDFS namespace is a chain of importance of records and files.

- Flat File Database

A level record database is a database which is secured on its host PC framework as a standard level report. To get to the structure of the data and control it, the report must be scrutinized totally into the PC's memory. Endless supply of the database operations, the document is again composed out completely to the host's record framework. In this put away mode the database is "level", which implies it has no structure for ordering and there are generally no basic connections between the records. A level document can be clarify content record [3].

- HIVE

The Apache Hive data software deals with large data. Managing large data-sets which will be present in distributed storage using SQL and database. Structure is molded into data already in storage which comes into HIVE. In HIVE we can make tables on the data we are getting from the previous platform just like in DATABASE and we can retrieve information from those tables ref <https://hive.apache.org/> Flume Flume Apache Flume is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of streaming data into the Hadoop Distributed File System (HDFS) [4].

- Spark Apache

Apache Spark is one of the big-data frameworks. Apache Spark is a fast and general engine for large-scale parallel data processing. It provides high-level APIs in Java, Scala, Python and R for data processing. I will be doing sentiment analysis in spark using Scala [5].

- Spark Streaming

Spark Streaming presents to Apache Spark's dialect coordinated API to stream handling, giving you a chance to compose spilling employments a similar way you compose group occupations. It underpins Java, Scala and Python. By running on Spark, Spark Streaming gives you a chance to use code for group data preparing, join streams against recorded information, or run specially appointed inquiries on stream state. Manufacture effective intuitive applications, not simply investigation. Spark Streaming can read information from HDFS, Flume, Kafka, Twitter and ZeroMQ [6].

Chapter 3

Requirement Specifications

3.1 Application Overview

Social media allows everyone to keep their fingers on the pulse of what's important that is going on, what's happening new around the globe and what's trending. To grow in their interests, what is trending will be a basic analysis of tweets, re-tweets or likes to develop an in-depth idea of the social consumer. We will be performing sentiment analysis on the data as well so that we can understand the attitude, meaning, opinions, likes, dislikes, emotions, hatred, love, sarcasm, positivity or negativity behind the words people have written. Sentiment analysis used in social media is of vital use because it gives us wider public opinion regarding certain topic and current affairs. This system will allow the users to have the insight of social media under one roof in which they are interested by entering a hashtag "#". Furthermore, we will perform sentiment analysis on that data, that is gathered. Basically, our system is marked to get data from social sites and perform sentiment analysis on the gathered data and displaying the dashboards on the results showing whether the HASHTAG we got is a positive, negative or neutral.

3.2 Application System Environment

We are using Oracle Big Data Lite Virtual machine which provide us incorporated environment to help us begin with the Oracle Big Data platform handling big data. As we are talking about getting data from twitter and Facebook that alone means lots of data downloading in milliseconds. Handling such large amount of data is one of the important factor. We are using spark for the sentiment analysis and Hadoop file system to save the downloaded data from twitter and Facebook.

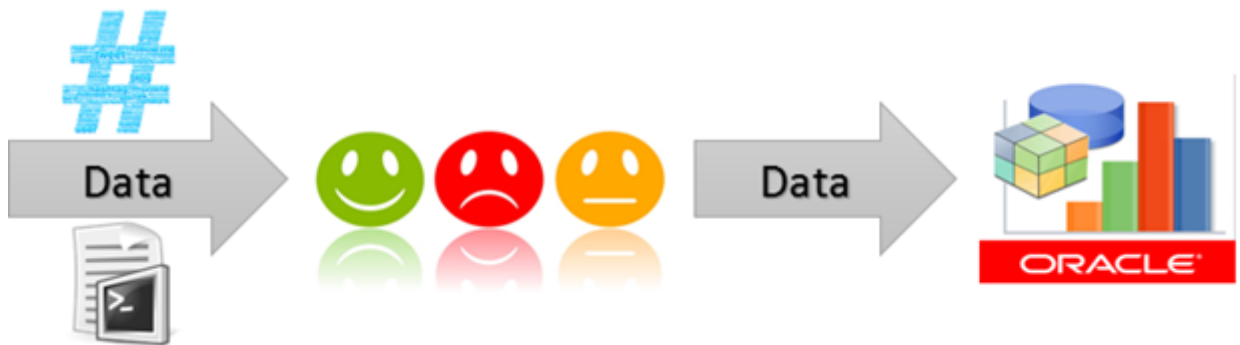


Figure 3.1: System Environment

3.3 Basic Functionality

Following are the basic functionalities provided by the system:

3.3.1 Twitter

The user will enter the hashtag and as soon as the user enter the hashtag and will click search, the tweets having that hashtag will start downloading form twitter. There will be a timer after that it will stop automatically. The download Jason file will be in memory and using spark streaming on the fly we will perform sentiment analysis on those Jason files to show that whether the downloaded tweets are positive, negative or neutral. After that those files will be saved on Hadoop/Hive in the form of tables. We will extract the data from hive tables and will display the dashboards boards showing the result.



Figure 3.2: Twitter Functionality

3.3.2 Facebook

Here the user has to hit the specific API of group. As there is a lot of privacy involved in Facebook so the user need the API Key of the specified group. As soon as the script written for the Facebook group is run you have to enter the group name and then the posts will start downloading. And rest of the functionality is same, will perform sentiment analysis and will display the dashboards.



Figure 3.3: Facebook Functionality

3.4 User Characteristics

Twitter: There is no user expertise involved. It is simple the user will enter the required hashtag. **Facebook:** We will be needing a person whose group's post we will be downloading because that person will have to provide us with the key which is only known to him and will expire after every 32 days. And a person who know how to generate and change the group API key in case we are supposed to get post from another group.

3.5 Functional Requirements

The following are the functional requirements of the system:

1. *Functional Requirement 1 (Computer/Laptop)*

It's an online system so the first thing we will be needing is the box to run the whole system.

2. *Functional Requirement 2 (Network)*

As soon as the hashtag is entered or the script is run the data starts downloading from twitter and Facebook so Internet is the most important entity we want for this project.

3. *Functional Requirement 3 (Twitter Access)*

As we are using Rest API's we need the twitter access for it.

4. *Functional Requirement 4 (Facebook Group)*

We need the Facebook account and a developer app to get the authenticated keys and rest API's.

5. *Functional Requirement 5 (HADOOP)*

As soon as the data start downloading the it goes directly to Hadoop file system and flume.

6. *Functional Requirement 6 (Sentiment)*

we will be performing sentiment analysis on the JSON files.

7. *Functional Requirement 7 (HIVE)*

Hive tables will be there and will be updating as soon as the new tweets/posts will be downloading.

8. *Functional Requirement 8 (Databases)*

To save the hive tables having tweets, posts and sentiment analysis's result

9. *Functional Requirement 8 (Dashboards)*

To show the result in the form of graphs and all using the BDD.

3.6 Non-Functional Requirements

The following are the non-functional requirements of the system:

1. *Security*

Twitter: As we are downloading the tweets from twitter we get the tweets which are public and no privacy is violated. The tweets which are not public will not be access by us.

Facebook: As security is the main preference for us so we will be downloading post from a specific group. The owner of that group will provide us his secret Key himself.

2. *Reliability*

Twitter: The system is reliable. The public tweets will start downloading the moment we send the hashtag **Facebook:** The post will start downloading as soon as the script start running, any updates in the post will be downloaded then.

3. *Timing*

People are tweeting and posting every second around the globe so downloading the tweets and posts should take less then milliseconds.

4. *Resilience*

Updates on the data will be handled properly as data is changing on web so fast. **Twitter:** when any tweet is re tweeted the table will update the number. **Facebook:** when someone comment on the post the new update extraction data will contain all the updates

5. *Capacity*

As we are handling tons of data there is huge capacity involved.

6. *Efficiency*

The system is efficient enough to download the tweets and re tweets and show their connection and can consume the load data utilization.

7. *Compatibility*

Our version of VM should be compatible to the spark we are using.

8. *Scalability*

This system can be used by any organization political or non-political to get the insight of their interest.

3.7 Hardware Selection

1. *HADOOP*

The software tool we using for extracting the data form different networks is HADOOP. It is an open source Framework. It cannot be installed because installation is very expensive

2. *Oracle Big Data Lite Virtual Machine*

Our system will run on this VM it provide us big data platform.

3. *System*

The minimum RAM of the system required for all the tools to run is 12.0 GB

4. *Scala*

The Scala Version should be 2.10.4 because it is the version supported and compatible with our VM.

3.8 Performance Requirements

All the system requirements should be fulfill to get the results of the analysis. The versions should be compatible with each other so they can perform the required task without failure. The RAM should be 12 GB or more to run the reporting in the end as well.

3.9 User Interface

There will be a web page containing two buttons

- Twitter
- Facebook

When we click on twitter the HASHTAG box will be open and when Facebook will be open the script will start running

- Sentiment analysis

This button will start the sentiment analysis of the downloaded tweets or posts and will generate the dashboards showing the graphs and results.

3.9.1 Tools

For displaying the dashboards and graphs we are using Big data discovery BDD tool where all the graphs dashboards and result will be displayed.

3.10 Use Cases

1. Main Use Case

The use case diagram shows the user interaction with the system.

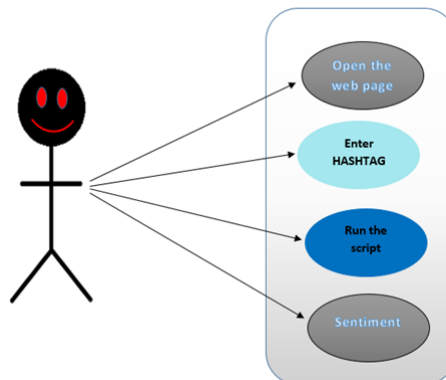


Figure 3.4: User and System Interaction

2. Use Case 1:

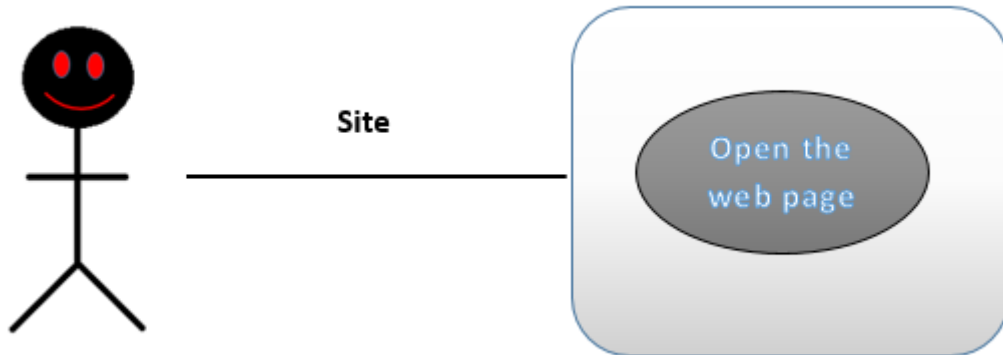


Figure 3.5: Opening the Webpage

Use Case ID	UC 1
Title	Open the Web page
Description	The user will open the web page on localhost
Primary Actor	User
Pre-Condition	There internet should be connected to the VM
Post-Condition	The web page will be displayed having an interactive GUI containing buttons of Twitter and Facebook

Table 3.1: Use Case 1

3. Use Case 2:

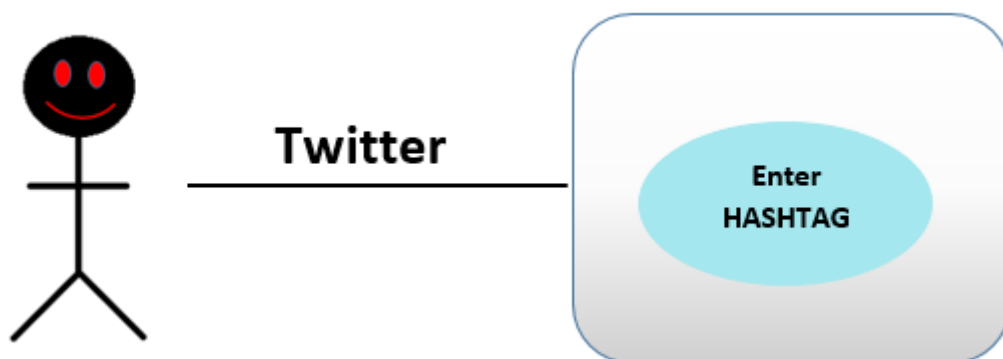


Figure 3.6: Twitter Access

Use Case ID	UC 2
Title	Enter HASHTAG
Description	Enter the # you want to search for on twitter or you want to know
Primary Actor	User
Pre-Condition	The user has to open the web page and then choose the option of TWITTER.
Post-Condition	The tweets containing that word will start downloading.

Table 3.2: Use Case 2

4. Use Case 3:

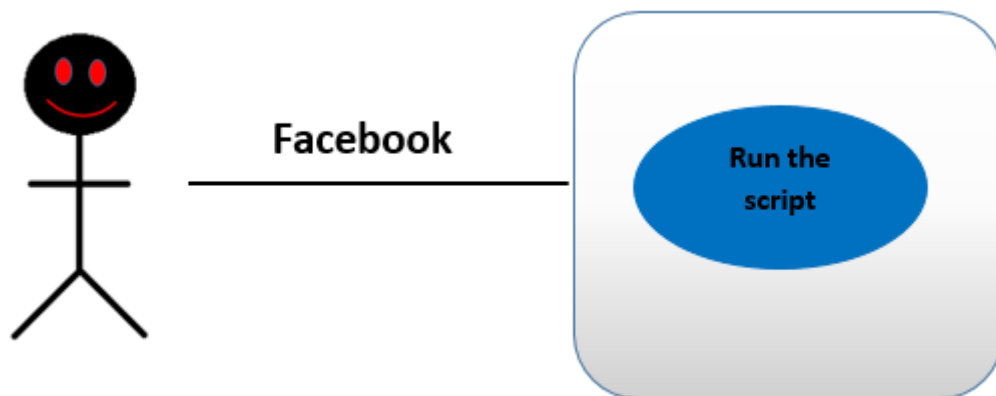


Figure 3.7: Facebook Access

Use Case ID	UC 3
Title	Run the Script
Description	You will click the button and script of Facebook enter the group name and it will start running
Primary Actor	User
Pre-Condition	The user has to open the web page and then choose the option of FACBOOK.
Post-Condition	The posts of the desired group will start downloading.

Table 3.3: Use Case 3

5. Use Case 4:

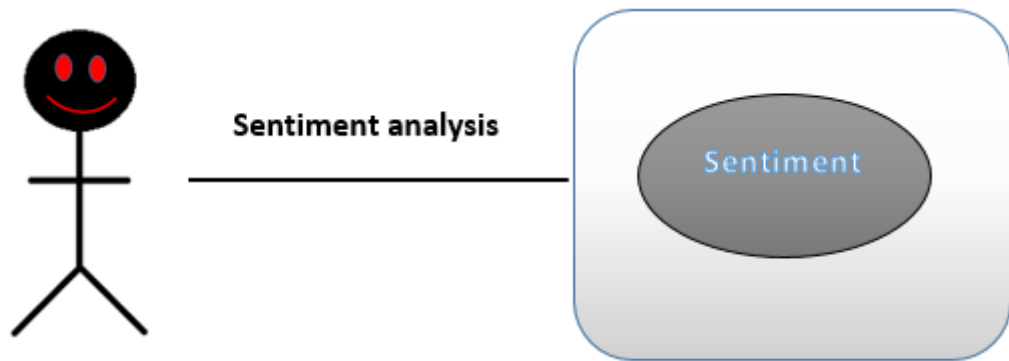


Figure 3.8: Sentiment Analysis

Use Case ID	UC 4
Title	Sentiment Analysis
Description	You will click the sentiment analysis button and the spark will start running on the JSON files.
Primary Actor	User
Pre-Condition	There should be Jason files present in the memory containing tweets or post downloaded by user via Twitter or Facebook respectively.
Post-Condition	The spark will start running sentiment analysis on the required file and will display positive, negative or neutral response.

Table 3.4: Use Case 4

Chapter 4

Design

The System Design describes what the system requirements, operating environment are and what kind of architecture and design can be built with those requirements.

4.1 System Architecture

We are using big data Virtual machine to build this whole setup. Our system include downloading data from two social sites i.e. Twitter and Facebook.

4.1.1 Twitter

We will enter the Hashtag and tweets containing that hashtag will start downloading.

4.1.2 Facebook

We will enter the Group name and the Facebook script will start running and downloading the posts.

4.1.3 Sentiment Analysis

The downloaded tweets or post will be in memory and spark streaming will do sentiment analysis on that data.

4.1.4 Dashboards

The dashboards will be displayed containing the result taken from Hive after sentiment analysis.

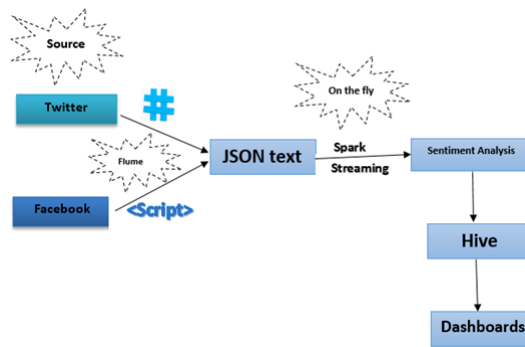


Figure 4.1: Describes the complete application architecture

4.2 Deployment Diagram

The following system deployment diagram shows how our system will be deployed on VM.

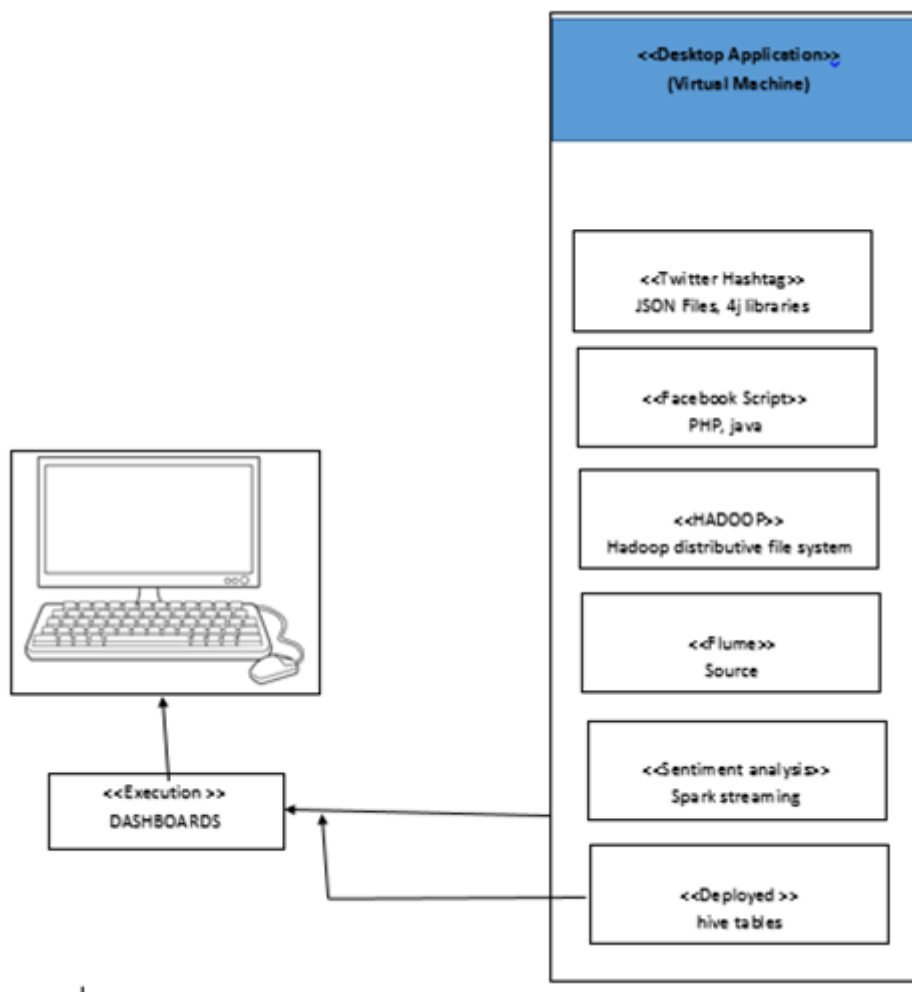


Figure 4.2: System Deployment Diagram

4.3 System Sequence Diagram

Following is the system sequence diagram:

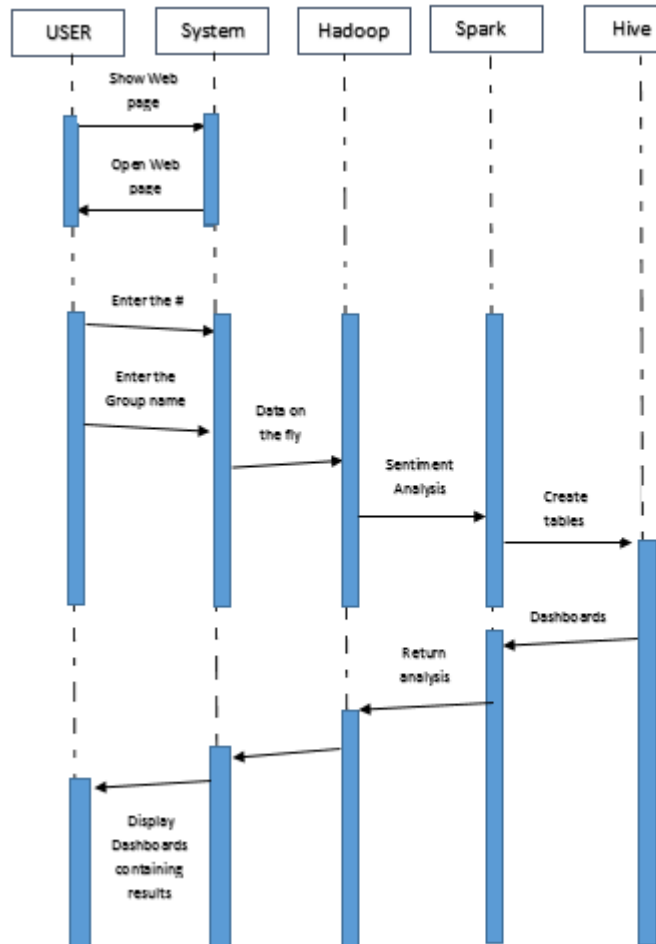


Figure 4.3: System Sequence Diagram

4.4 High-level Diagram

Following is the system high-level diagram:

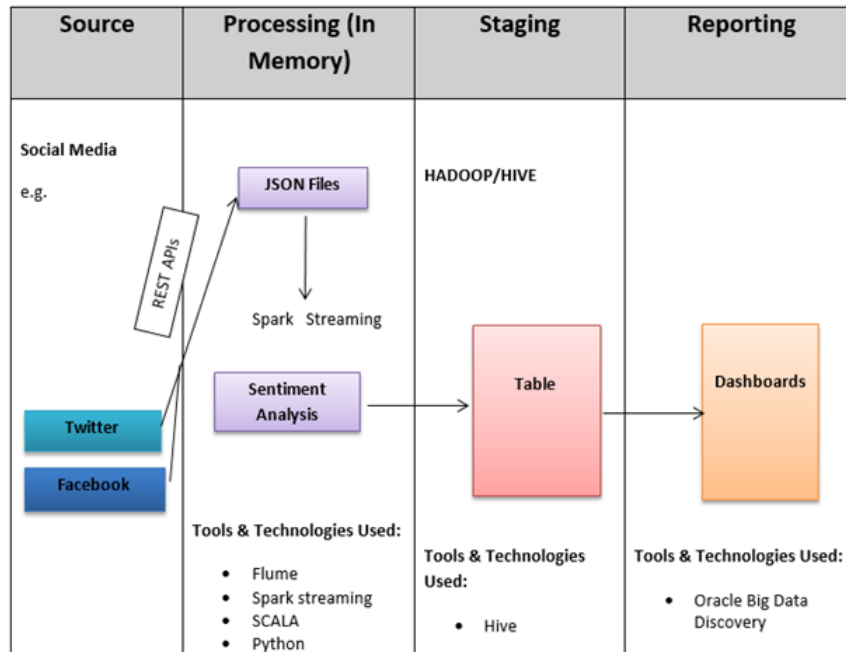


Figure 4.4: High-level Diagram

4.5 Process Interaction Models

The following are the process models:

1. *Process Model 1:*



Figure 4.5: Twitter Process Model

Process ID	P 1
Title	Twitter Process
Description	Tweets downloading in the form of JSON
Primary Actor	User
Pre-Condition	Internet connection and web-page access via localhost
Post-Condition	Tweets having the respective hashtag will be downloading.

Table 4.1: Process Model 1

2. Process Model 2:



Figure 4.6: Facebook Process Model

Process ID	P 2
Title	Facebook Process
Description	The Facebook group script will start running and post will start downloading.
Primary Actor	User
Pre-Condition	Internet connection and web-page access via localhost
Post-Condition	Posts of the group will be downloaded.

Table 4.2: Process Model 2

3. Process Model 3:



Figure 4.7: Sentiment Analysis Process Model

Process ID	P 3
Title	Sentiment Analysis Process
Description	Spark streaming will perform Sentiment Analysis on the downloaded file (Facebook or tweets)
Primary Actor	User
Pre-Condition	Internet connection and web-page access via localhost. The file should be downloaded containing tweets or posts.
Post-Condition	Information will be saved in Hive tables and dashboards will be displayed.

Table 4.3: Process Model 3

Chapter 5

System Implementation

In this chapter, I will discuss the tools, technologies and software's I will be using for my project

5.1 Tools and Technologies

5.1.1 Virtual Machine

Oracle Big Data Lite Virtual Machine gives an incorporated situation to help you begin with the Oracle Big Data stage. I'm using this Virtual machine to build the whole environment for my system. Oracle Big Data platform which includes the Cloudera Distribution of Apache Hadoop

5.1.2 Big Data

Social media contains all kind of data, large data which comes under the term "Big data". Keeping in mind today's world the concept of social sites is growing and with that Big data is also growing and its impact in near future will be more important as Big Data is today, its impact will be even more notable. The reason behind this is the Big data itself. It defines the whole generic set of data. The following diagram explains the terms which comes under Big data



Figure 5.1: 4V's of Big Data

5.1.3 HADOOP

HADOOP is software frame work that process huge measure of information sets in a conveying figuring environment and bunches of PCs. It is an environment and all the tools are present on that environment. The Apache Hadoop programming library is a structure that takes into account the conveyed handling of expansive information sets crosswise over bunches of PCs utilizing basic programming models. Hadoop Project develops open-source programming for solid, versatile, dispersed computing. It is intended to scale up from single servers to a large number of machines, every offering neighborhood calculation and capacity, instead of depending on equipment to convey high-accessibility, the library itself is intended to distinguish and handle failure at the application layer, so conveying a profoundly accessible administration on top of a bunch of PCs. Apache Hadoop is the industry-standard open-source Big Data Platform for distributed storage and distributed batch processing of Big Data. Also, the Hadoop project incorporates a family of tools also known as "Hadoop Ecosystem" to address particular needs [2].

5.1.4 HADOOP Distributed File System *HDFS*

HDFS is distributed file system which gives us high access to application data we are dealing with. HDFS is a capacity framework utilized by HADOOP applications. It is the file system of HADOOP and is exceedingly fault tolerant and is intended to be sent on minimal effort equipment.

5.1.5 HIVE

The Apache Hive data software deals with large data. Managing large data-sets which will be present in distributed storage using SQL and database. Structure is molded into data already in storage which comes into HIVE. In HIVE we can make tables on the data we are getting from the previous platform just like in DATABASE and we can retrieve information from those tables.

5.1.6 Flume

Apache Flume is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of streaming data into the Hadoop Distributed File System (HDFS) [4].

5.1.7 Spark

Apache Spark is one of the big-data frameworks. Apache Spark is a fast and general engine for large-scale parallel data processing. It provides high-level APIs in Java, Scala, Python and R for data processing. I will be doing sentiment analysis in spark using Scala [5].

5.1.8 Big Data Discovery

It is a tool provided by Apache Hadoop and oracle VM. It makes online dashboards on running data taken from the HIVE [?].

5.1.9 Oracle Data Integrator *ODI*

Oracle Data Integrator is a far reaching information combination stage that covers all information incorporation prerequisites: from high-volume, superior bunch burdens, to occasion driven, stream bolster coordination procedures, to SOA-empowered information administrations. Oracle Data Integrator *ODI* 12c, the most recent rendition of Oracle's vital Data Integration offering, gives unrivaled designer profitability and enhanced client involvement with an overhauled stream based user interface and data manipulation. It is designed with the superior engineering with extensive enormous big data and included parallelism when executing information incorporation forms [7].

5.2 Languages

I am using different languages for different purposes. I will be writing scripts and code in PHP, Python, for sentiment analysis I'll be using Scala and for retrieving data, Scala is a modern multi-paradigm programming language, by multi-paradigm it means that it incorporates multiple programming models: Object-Oriented, Functional, Statically typed and extendible then the data will be saved in HIVE table I'll be using HQL and generating the dashboards.

5.3 Methodology

I will be downloading data from social media i.e. Twitter and Facebook. I will be downloading data from social media i.e. Twitter and Facebook. The process will be as

follow We will open the web page from local host. The web page contains two tabs one for Twitter and one for Facebook. When the user will click Twitter tab the new Twitter page will be generated. There user will enter the 'Hashtag'. As soon as the 'Hashtag' is enter the Flume will update the keyword in configuration file i.e. the 'hashtag' and tweets will start downloading. The time out of flume is 4 minutes. The flume will automatically stop downloading tweets after 4 minutes. The flume will dump those tweets on port from where Spark will listen. After every 5 minutes the spark will come to the port and will listen for the tweets the moment it will get the tweets it will start running the code of spark streaming and sentiment analysis will be performed on those tweets. After doing sentiment analysis the spark will dump the result in Hive in the form of external tables, from there we will extract the data to normal table and will generate the ODI packages behind which the scripts of tweets and retweets will be running and we will generate the live dashboards by logging in to oracle Big Data Discovery BDD. The flume will take 1 second to update the configuration file. Spark take 9 – 10 minutes to listen and perform sentiment analysis. It is sometimes really slow because of the internet connection. The large servers run the flume, spark, Dashboards without any delay, within few minutes. The methodology is same for the Facebook apart in Facebook we are doing the same process for group and page.

5.3.1 Facebook

There are 3 kind of group's public, private, closed. You can get the token of a public group and can access the posts which are public, but only if the group is public *i.e. thegroup'sprivacysettingisOPEN* If the group is private you'll be needing the `user_managed_groups` permission to read the group content from the admin of the group. You'll request the admin for token and he will grant you the permission by giving you the required token, which only admin can generate.

- Process

First of all I will create an App by going to developer's site provided by Facebook

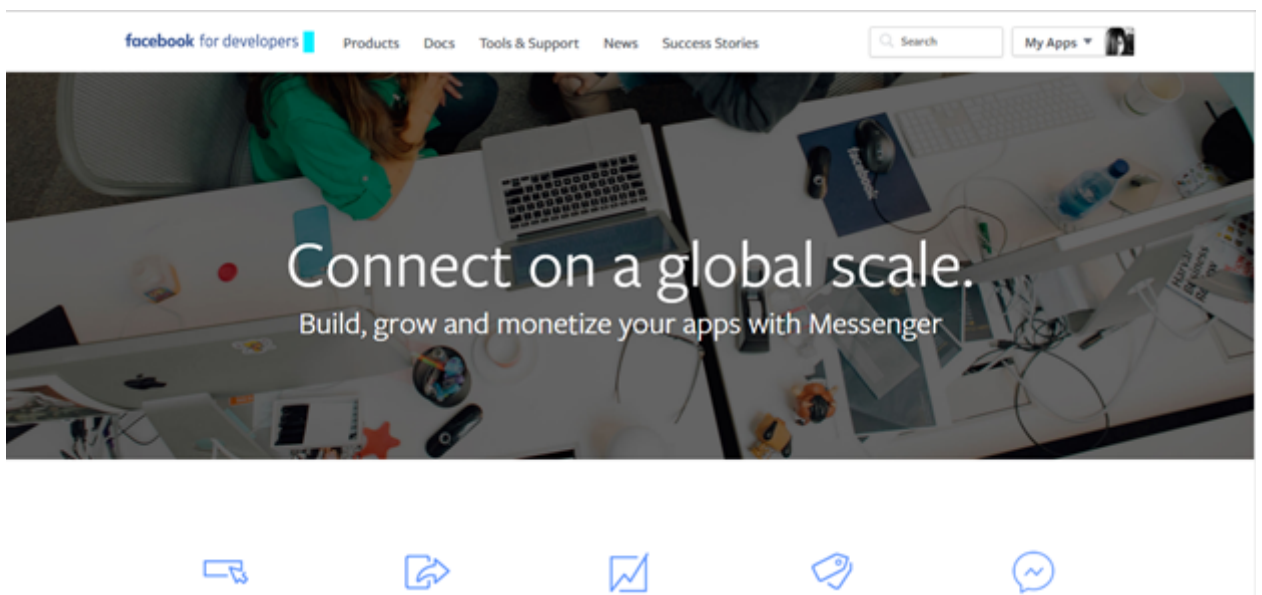


Figure 5.2: Facebook Developer Site

Following are the steps:

1. Login to Facebook
2. Create Developer Account
3. Create new Facebook app
4. Choose Platform
5. Choose a Name
6. Follow "Quick Start" Steps
7. Generate App ID tokens
8. Protect your App Secret

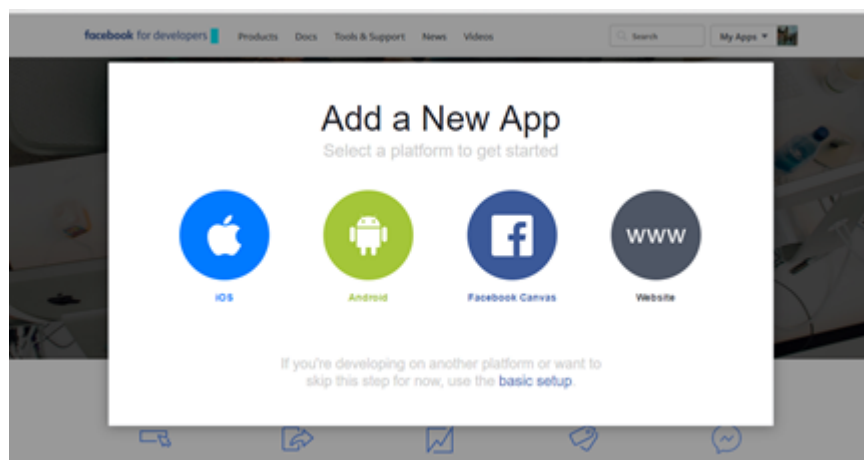


Figure 5.3: Facebook App: Project

Give the basic information the app's name and your own email I-D and **Create App ID**.

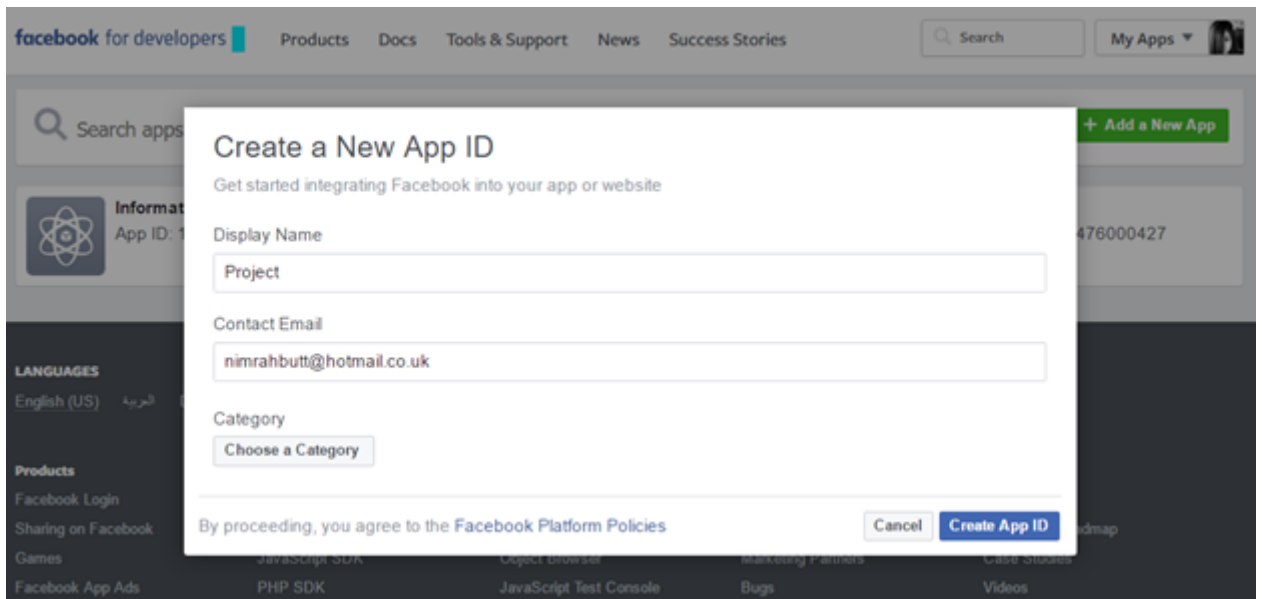


Figure 5.4: Facebook App: Application ID

Before generating the App ID it will take your security test and after that it will create an App ID for me which I will be using throughout my project to get data via this app.

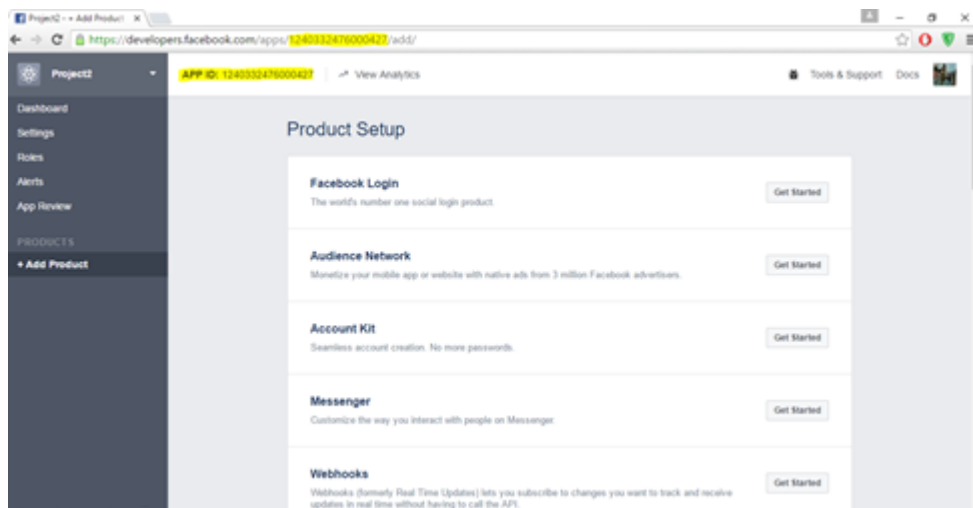


Figure 5.5: Facebook App: Product Setup

Then I'll generate the App Secret that is the token we going to use for downloading the data

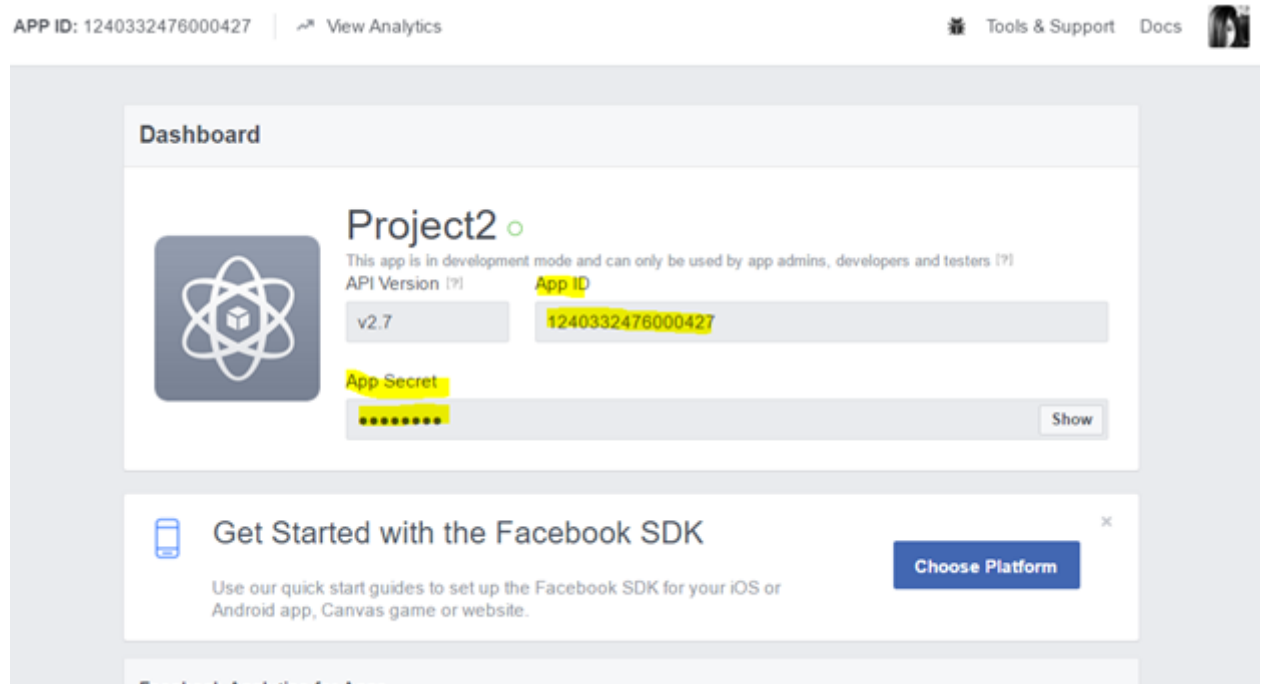


Figure 5.6: Facebook App: Token Generation

When we will choose the option “show” it will ask the password of my account, which means that the token is secure to my ID, and token will remain secret unless or until I gave it to someone.

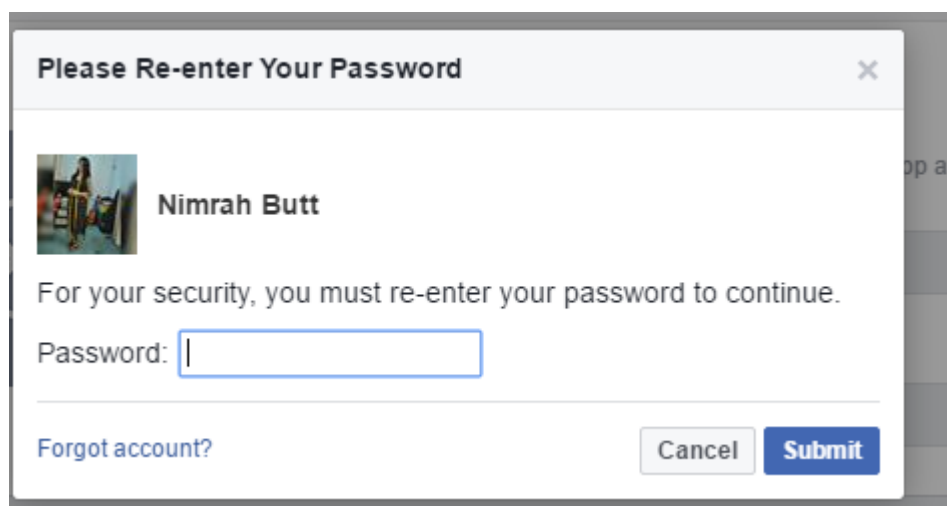


Figure 5.7: Facebook App: Application Security

There are two types of token:

1. Short lived

Short term tokens will expire after 3 days of generation. And when you'll enter the old token it will display this message token has expired.

2. Long lived

Long term tokens will expire after 60 days of generation. I'll be using the long term tokens. Once I get the token, I will write the scripts for it.

- Working on Virtual Machine

We will run the Virtual Machine



Figure 5.8: Running the Virtual Machine

I'll login to oracle VM to get access of it. I'll connect to the server. First of all I'll make the directory where the data will be stored. The data which I'll be downloading from Facebook will come to flume and will be stored there, for that first of all I have to make a directory in the flume and that path will be given to the script to download data there.

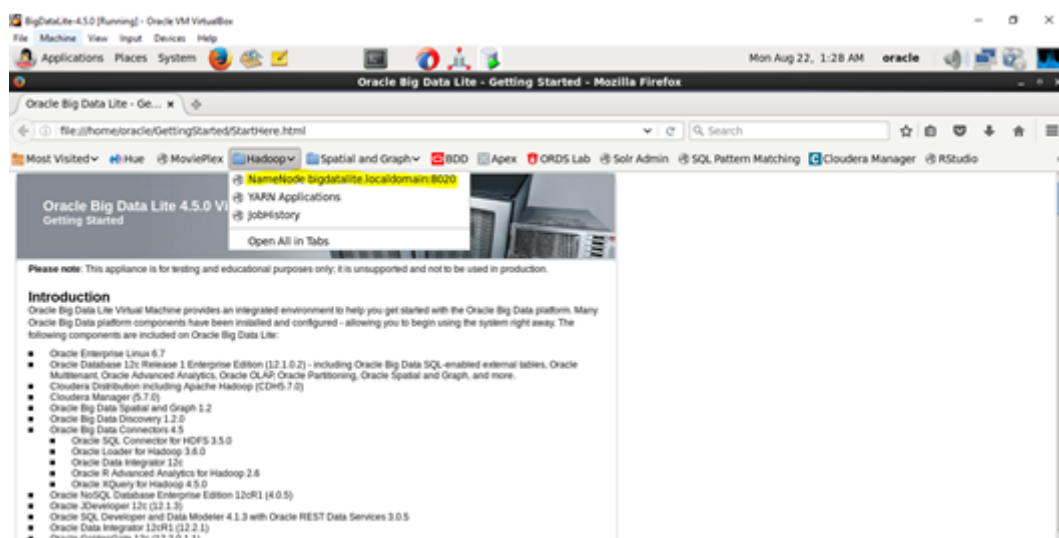


Figure 5.9: Oracle Big Data Lite



Figure 5.10: Oracle Big Data Lite: Local Domain

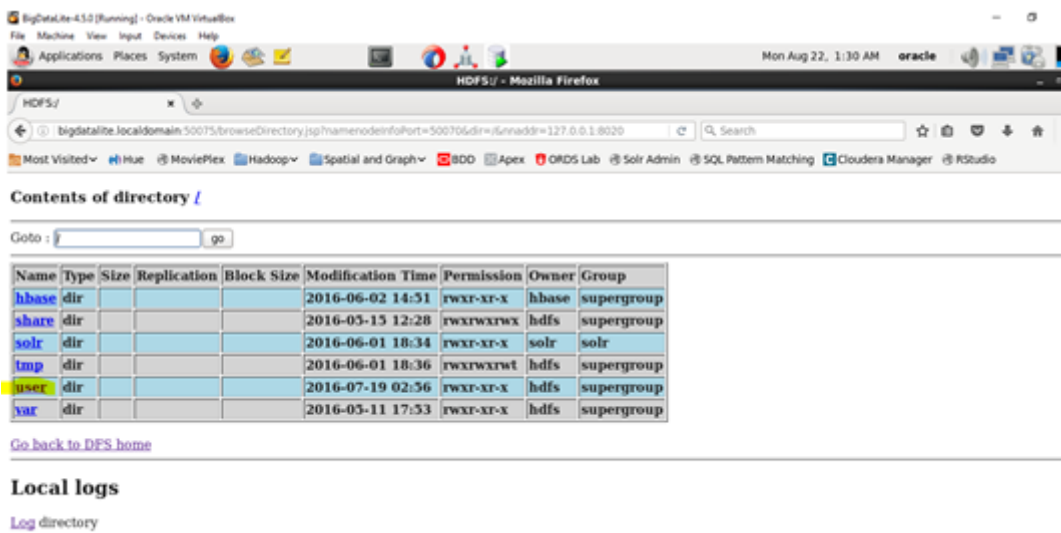


Figure 5.11: Oracle Big Data Lite: Content of Directory

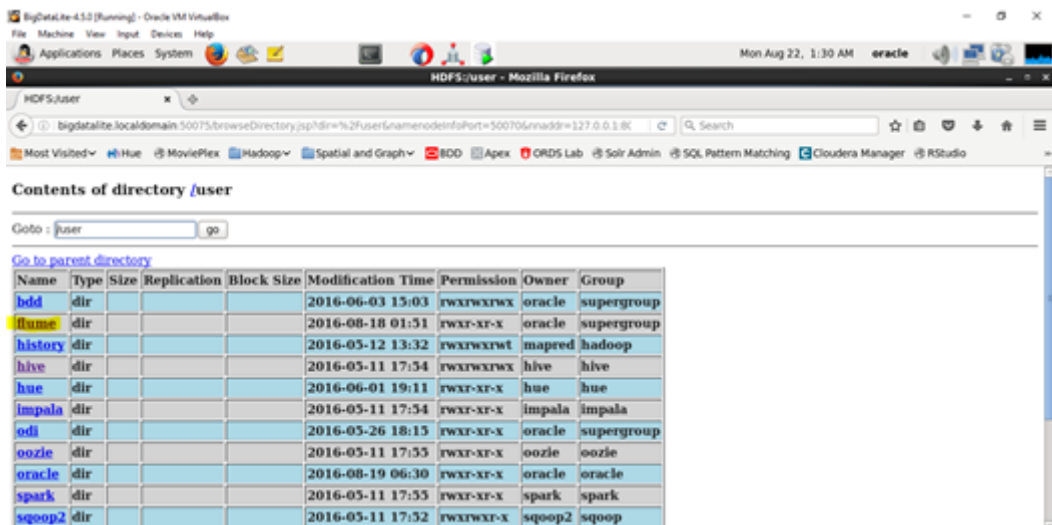


Figure 5.12: Oracle Big Data Lite: Flume

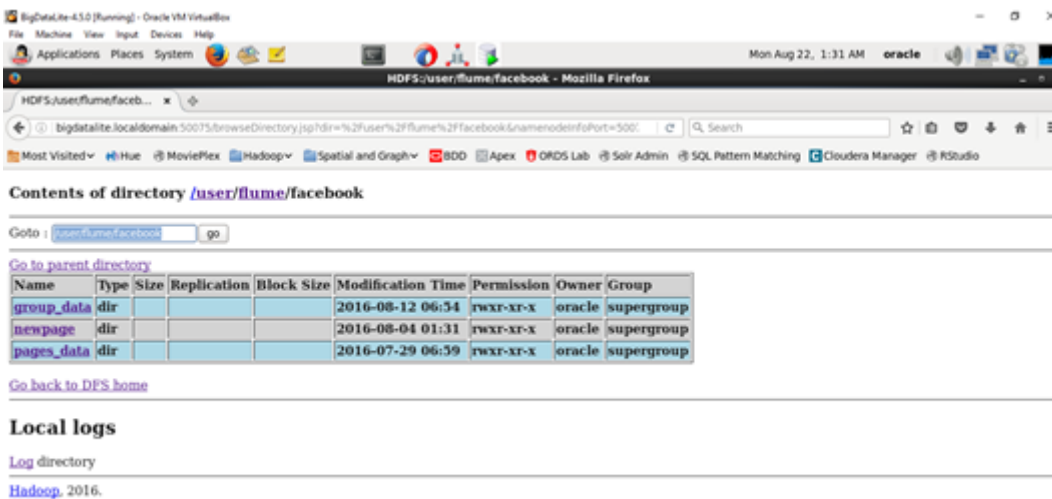


Figure 5.13: Directories

Here all the umes les will be downloaded Path: /user/ume/facebook These are the umes downloaded les Now I'll write scripts for it. First of all I'll made a ume conguration le. In file configuration file there I'll make a flume agent and will name it Facebook agent.

- Flume Agent:

An agent is a very important process in Flume. The job of agent is to receives the data from clients or agents and forwards it to its next place/ sink/workplace/agent.

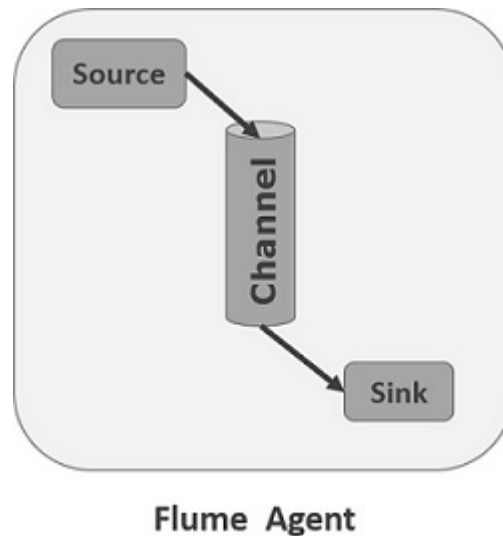


Figure 5.14: Flume Agent

Basically the system needs:

- A java run-time environment
- Memory - Sufficient memory for configurations used by sources, channels or sinks
- Disk Space - Sufficient disk space for configurations used by channels or sinks
- Directory Permissions - Read/Write permissions for directories used by agent.

I've already created the java run-time environment so now I'll make the rest. I'll congure the source and will give path of the shell script. Furthermore, I'll dene the sink which includes type, channel, hostname, port, HDFS channel, type, path, le type, write format, batch size, roll size, roll count, going further I'll dene a channel which buffer the event in memory *capacity* and then I'll bind the source and sink to the channel.

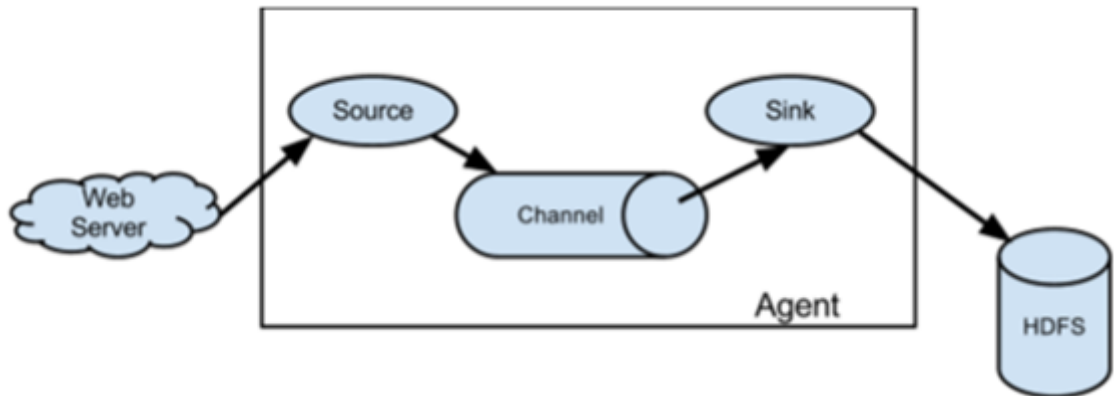


Figure 5.15: Sources and Sinks

```

facebook_groupconf - Notepad
File Edit Format View Help
# example.conf: A single-node Flume configuration

# Name the components on this agent
FacebookAgent.sources = r1
FacebookAgent.sinks = HDFS
FacebookAgent.channels = c1

# Describe/configure the source
FacebookAgent.sources.r1.type = exec
FacebookAgent.sources.r1.command = sh /home/oracle/Desktop/facebook_script.sh

# Describe the sink
#FacebookAgent.sinks.k1.type = avro
#FacebookAgent.sinks.k1.channel = c1
#FacebookAgent.sinks.k1.hostname = localhost
#FacebookAgent.sinks.k1.port = 44444
FacebookAgent.sinks.HDFS.channel = MemChannel
FacebookAgent.sinks.HDFS.type = hdfs
FacebookAgent.sinks.HDFS.hdfs.path = hdfs://127.0.0.1:8020/user/flume/facebook/group_data
FacebookAgent.sinks.HDFS.hdfs.fileType = DataStream
FacebookAgent.sinks.HDFS.hdfs.writeFormat = Text
FacebookAgent.sinks.HDFS.hdfs.batchSize = 1000
FacebookAgent.sinks.HDFS.hdfs.rollSize = 0
FacebookAgent.sinks.HDFS.hdfs.rollCount = 10000

# Use a channel which buffers events in memory
FacebookAgent.channels.c1.type = memory
FacebookAgent.channels.c1.capacity = 10000
FacebookAgent.channels.c1.transactionCapacity = 10000

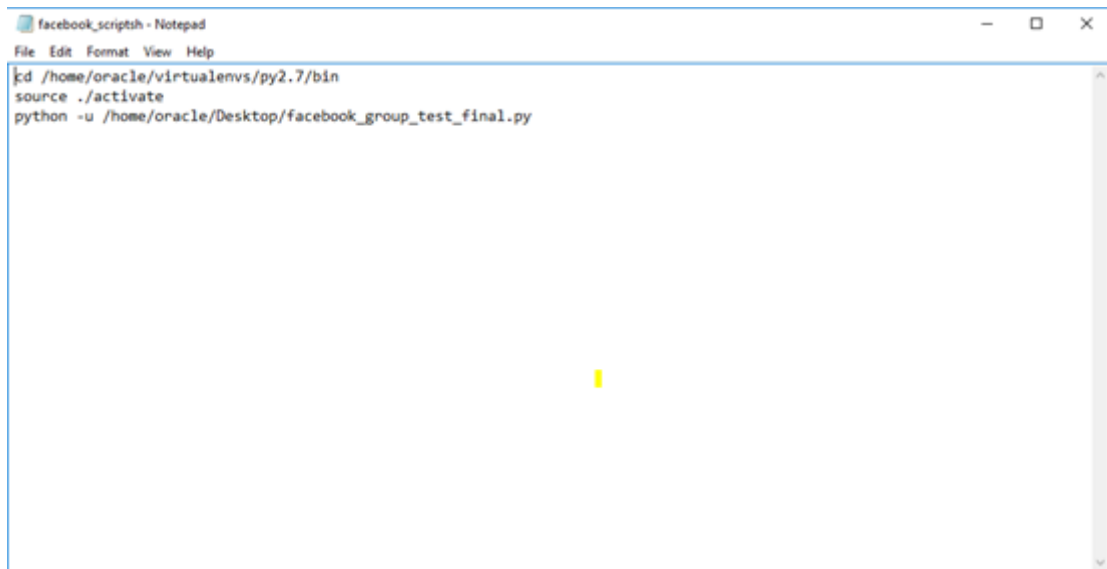
# Bind the source and sink to the channel
FacebookAgent.sources.r1.channels = c1
FacebookAgent.sinks.HDFS.channel = c1
  
```

Figure 5.16: Facebook Group Configuration

- Shell Script

Then the path of the shell script which was given in the configuration file will run. I'm using a 4.5.0 version of the VM which supports all the libraries of Twitter but doesn't support some sink libraries of Facebook, so instead of installing the latest version of the VM, I'll create a virtual environment of the latest VM. So when the shell script will run on the terminal, it

of all it will create a virtual environment and will activate it and further it'll be executed on python.

A screenshot of a Notepad window titled "facebook_script.sh - Notepad". The window contains three lines of shell script code:

```
cd /home/oracle/virtualenvs/py2.7/bin
source ./activate
python -u /home/oracle/Desktop/facebook_group_test_final.py
```

Figure 5.17: Facebook Shell Script

- Creating Python Script

Will import all the Jason files and libraries. It will take the token the same token which we generated using the facebook app. and will run the script and print the tuples we wanted to print. We have used loops to print the post and the comments on that posts. There is also another thing called story i.e. if the post has shared some a page link. I have pipelined it so it will give us in squence and only one script can be used. I'll be using end points of the fabeook API to get the data. The most important thing here is the base url we will using that is of facebook. Followed by the access token. We'll write down the code here accessing the post and comments.



```

facebook_group_test_final.py - Notepad
File Edit Format View Help
import json
import os
import urllib2
import requests
import time
#import urllib.request

def return_data(url, api_key):
    request = urllib2.Request(url)
    request.add_header('Authorization', 'Bearer {}'.format(api_key))
    # print(request)
    response = urllib2.urlopen(request)

    # encoding = response.headers.get_content_charset()

    data = json.loads(response.read())

    return data

facebook_api_key = ""
def main():
    # base_url_oauth = "https://graph.facebook.com/v2.7/oauth/access_token"

    a = 0
    # data_oauth = return_data(base_url_oauth+"?client_id=1651805001718251&client_secret=a564ac76ba62b451798e8f1b853893ae&grant_type=client_credentials", a)
    # print json.dumps(data)
    # facebook_api_key = data_oauth["access_token"]
    facebook_api_key =
'EAA8cTC9E9ng0BAH6ntZIFJKK4gzf1Uhtg1Lj3gYoeTFranRPM6VE3ZBKoEEv8F2WaeV5yPdSpjzEKMaSX4Mra3V1d38nGXZA0KzIPFvcSektQsVZBJYQsh1ZAAZAPy1lqly5KTUKyJGZAIr21Wsl61cKaeYrEHcwsZD'

    base_url_id = "https://graph.facebook.com/search?q="
    nextUrl_id = base_url_id + "BigDataTimeline&type=group"
    data_id = return_data(nextUrl_id, facebook_api_key)
    # print json.dumps(data_id["data"][0]["id"])
    data_id = data_id["data"][0]["id"]
    if data_id.startswith('') and data_id.endswith(''):

```

Figure 5.18: Facebook Group Test

In .sh file I'll take the input from users. i.e. which page data they need *which I have already coded*. In my case as I'm taking specific group whose token I've already generated so I'll have to type that. I'll start from taking the name from user I'll get the access token by combining two URLs. First one will be the base URL of the group which will be like this: https://graph.facebook.com/v2.7/oauth/access_token? the other part will be data URL like this:

```
client_id=1651805001718251&client_secret=a564ac76ba62b451798e8f1b853893ae&grant_type=client_credentials
```

Combining these two will give you the access token (either short term or long term) (for Facebook group). I'll simply copy paste the key provided by the admin of the group as facebook_api_key I'll type the base URL the id from which a group can be searched which will be saved in base_url_id like the following command:

```
base_url_id = \"https://graph.facebook.com/search?q=\"
```

Now it will go to facebook and will search the given URL and the group once it will hit that it will return the id and the api key provided by the admin followed and combined by the DATA ID and feed. Feed is the post of a group. It will start printing the data of the group in one hit. Usually when all the post of first page is accessed there is a URL of the next page in the end of that page. We have to click the URL to go to the next page access

the posts. We'll be doing paging here which will ease up the work and instead of url it will keep loading the post of next pages as soon as the first page's post are finished. We'll also provide the path of the python file.

After writing all these code. I'll go to Vm and first of all will create the virtual environment.

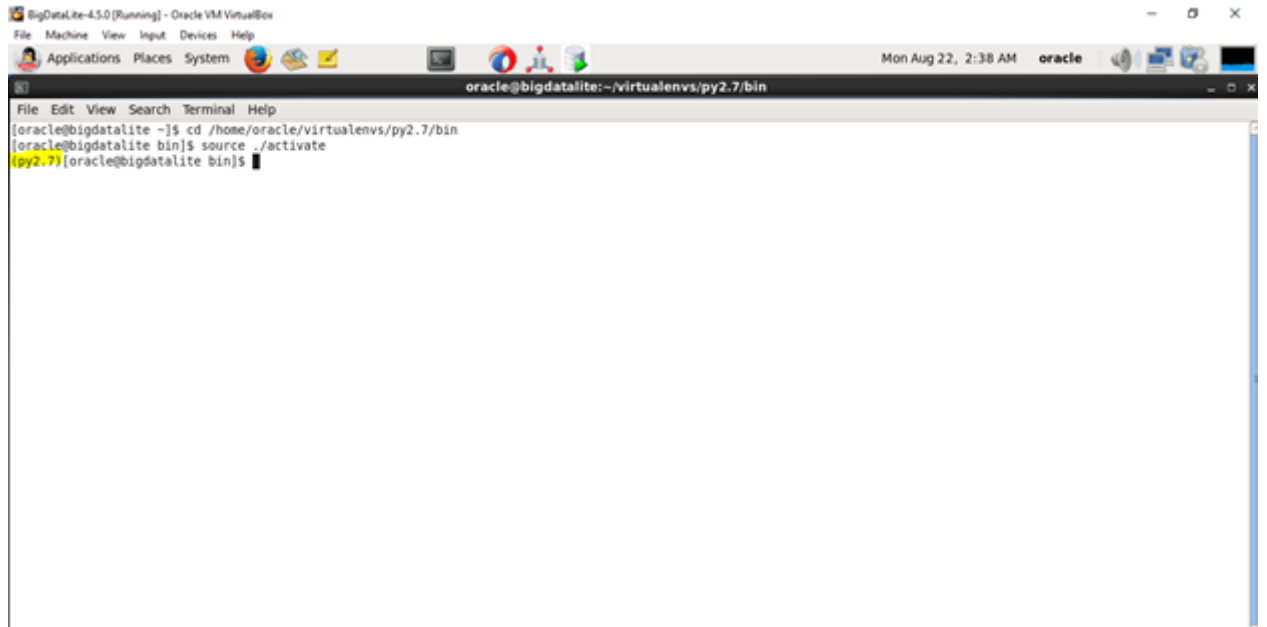


Figure 5.19: Oracle VM

Now I'll run the shell script of group that I've written. Before running the script the facebook_group folder we created is empty because there is no data in the flume. These are the flumes downloaded files

[Go to parent directory](#)

Name	Type	Size	Replication	Block Size	Modification Time	Permission	Owner	Group
FlumeData.1470136223325	file	1.64 KB	1	64 MB	2016-08-02 07:10	rw-r--r--	oracle	supergroup
FlumeData.1470285526246	file	1.64 KB	1	64 MB	2016-08-04 00:39	rw-r--r--	oracle	supergroup
FlumeData.1470288648276	file	1.64 KB	1	64 MB	2016-08-04 01:31	rw-r--r--	oracle	supergroup
FlumeData.1471848681010	file	1.64 KB	1	64 MB	2016-08-22 02:51	rw-r--r--	oracle	supergroup
FlumeData.1471848991524	file	1.64 KB	1	64 MB	2016-08-22 02:56	rw-r--r--	oracle	supergroup

[Go back to DFS home](#)

Figure 5.20: Flume Downloaded Files

Now the flume will through this data to the port where spark will listen the port and will take data for sentiment analysis *explained later*

5.3.2 Twitter

Twitter is an online social networking service that enables users to send and read short 140-character messages called tweets. It has around 310M Monthly active users and around 100million daily active users from countries all over the world. As of January 2016 there were reported an average of over 300million tweets per day. Twitter is a simple usage model and availability of a wide range of APIs to access raw content through external systems making it more developer friendly platform for various applications. Twitter keeps a track of "trending" topics of conversations across geographical boundaries and time by "indexing" tweets by keywords popularly known as hashtags and lists public tweets. Using the Twitter Public Streaming API, external systems can ingest public tweets being generated by public users in real-time and also filter them using a specific keyword or hashtag of interest to the system. Twitter allows you to download its data i.e. Tweets and Re-tweets by using twitter streaming API's. I'm using the server side scripting language

PHP,python to make requests to twitter API's to get the tweets and re-tweets about specific hashtag the user has entered. The tweets and re-tweets will be downloaded in JSON format that is easily readable. As soon as the Hashtag is written there is a certain Twitter 4j libraries which will hit the Hashtag's API and get the result to us we send that data to spark streaming and there it perform sentiment analysis and we then store that result in HDFS i.e. HADOOP distributed file system.

- Process

First of all I will be making app on twitter to get the token and API's for the project to run the les.

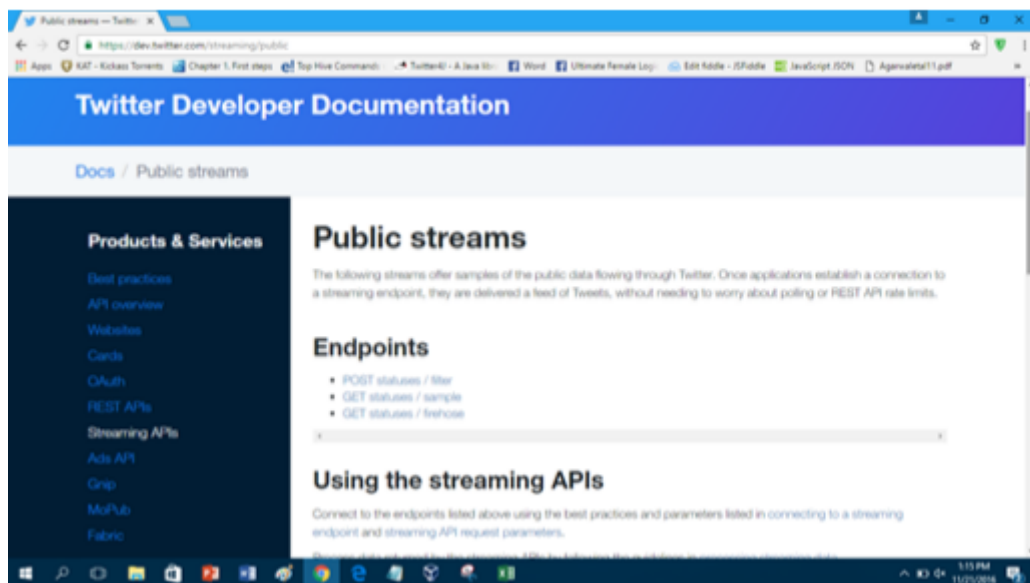


Figure 5.21: Twitter Developer Documentation

In twitter there is one jar file I'll be making and placing it in the specified flume folder. I'll set the java home.

I'll login to oracle VM to get access of it same like I did for Facebook. I'll connect to the server. First of all I'll make the directory where the tweets will be stored. The data which I'll be downloading from Twitter will come to flume and will be stored there, for that first of all I have to make a directory in the flume and that path will be given to the script to download data there.

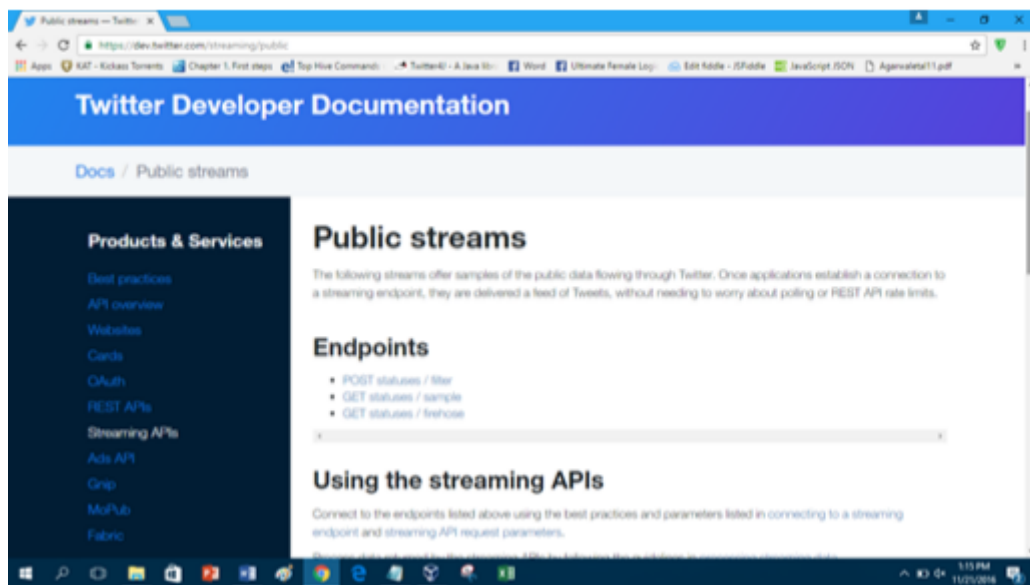


Figure 5.22: Twitter Developer Documentation

In twitter there is one jar file I'll be making and placing it in the specified flume folder. I'll set the java home. I'll login to oracle VM to get access of it same like I did for Facebook. I'll connect to the server. First of all I'll make the directory where the tweets will be stored. The twitter recently changed all of its end point through which we access the tweets, so for that purpose we need to configure the files and update them by extracting the new end points by hitting the API of Twitter and created new end point which download tweets from twitter further allow to perform sentiment analysis by spark As I have changed all the end points of API's which I have hit to the user end points so now tweets will start downloading. The data which I'll be downloading from Twitter will come to flume and will be stored there, for that first of all I have to make a directory in the flume and that path will be given to the script to download data there.

- FLUME.CONF File

There will be a flume configuration file. Which will contain all the information how we will connect the twitter to the download the tweets flumes. I'll make a variable

named Twitteragent, just like I did for Facebook. In flume agent Twitter is source Flume is agent HDFS is sink Now It will perform the same task as Facebook agent. The hashtag the user will enter will automatically update the “keyword” in ume configuration le. The tweets will be in the JSON format. The flume will place these tweets on Memory and from their spark will listen to the same port on which flume is throwing the tweets and spark will perform sentiment analysis on these les using spark stream. The hashtag the user will enter will automatically update the “keyword” in ume configuration le. The tweets will be start downloading in JASON format after 4 minutes the tweets will be stop downloading and flume will place those les port 44444 and spark will take charge from there it will listen to the port after every 5 seconds, take the tweets and will perform sentiment analysis on these les using spark stream. This is the port on which flume is dumping the tweets and spark is listening to.

```
# Licensed to the Apache Software Foundation (ASF) under one
# or more contributor license agreements. See the NOTICE file
# distributed with this work for additional information
# regarding copyright ownership. The ASF licenses this file
# to you under the Apache License, Version 2.0 (the
# "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing,
# software distributed under the License is distributed on an
# "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
# KIND, either express or implied. See the License for the
# specific language governing permissions and limitations
# under the license.
# The configuration file needs to define the sources,
# the channels and the sinks.
# Sources, channels and sinks are defined per agent,
# in this case called 'TwitterAgent'
TwitterAgent.sources = r1
TwitterAgent.channels = c1
TwitterAgent.sinks = k1
TwitterAgent.sinks.k1.type = avro
TwitterAgent.sinks.k1.channel = c1
TwitterAgent.sinks.k1.hostname = localhost
TwitterAgent.sinks.k1.port = 44444
TwitterAgent.sources.r1.type = com.cloudera.flume.source.TwitterSource
TwitterAgent.sources.r1.channels = MemChannel
TwitterAgent.sources.r1.consumerKey = 10zDuIEr7eD0niXYUQSLq1IVP
TwitterAgent.sources.r1.consumerSecret = zHHxBvOLpZe787wbFEY6SMzHV9Ewisa2YZnDGV2S11PvZrEWQb
TwitterAgent.sources.r1.accessToken = 3001083326-Axctur6DVGmFQcREVTparPqxLp5mq2K0j3jeiG1
TwitterAgent.sources.r1.accessTokenSecret = mxihKq24jwPeZYuKWdYPrldczxXTJ67WUsmqJr3UWz00p
TwitterAgent.sources.r1.keywords = yasirshah
TwitterAgent.channels.c1.type = memory
TwitterAgent.channels.c1.capacity = 1000
TwitterAgent.channels.c1.transactionCapacity = 100
```

Figure 5.23: Flume Configuration


```

TwitterAgent.sources.r1.channels = MemChannel
TwitterAgent.sources.r1.consumerKey = 10zDuIEr7eD0niXYUQSLq1IVP
TwitterAgent.sources.r1.consumerSecret = zHHxBv0LpZe787wbfEY6SMzHV9Ewisa2YZnDGV2S11PvZrEWQb
TwitterAgent.sources.r1.accessToken = 3001083326-Axctur6DVGmFQcREVTparPqxLp5mq2K0j3jeiG1
TwitterAgent.sources.r1.accessTokenSecret = mxiiWkq24jwPeZYuKWdYPrldczxXTJ67WUsmqJr3UWz0Op
TwitterAgent.sources.r1.keywords = #yasirshah
TwitterAgent.channels.c1.type = memory
TwitterAgent.channels.c1.capacity = 1000
TwitterAgent.channels.c1.transactionCapacity = 100

```

Figure 5.24: Twitter Agent

5.3.3 Sentiment Analysis

Sentiment Analysis in simple terms, is the task of evaluating and classifying the mood/view of the user w.r.t the semantics/meaning of a piece of unstructured or semi-structured content generated by a user of a system into a quantifiable/measurable metric either categorical or numerical in format. The former categorical or nominal mapping of semi-structured data such as that of Twitter and Facebook, directly delivers a useful information for many Business Analytics applications.

- Process

We will be doing sentiment analysis using the spark **Scala**: Spark Programming Language Every Spark application consists of a driver program that runs the user's main function and executes various parallel operations on a cluster. The cluster can be any cluster in our case the HADOOP DFS Cluster.

Basic Operations: All data operations and transformations in Spark are performed on the Sparks abstraction of an immutable distributed data-set called RDD (Resilient Distributed Dataset). RDD is a fault-tolerant collection of elements that are partitioned across the nodes of the cluster that can be operated on in parallel. Basic RDD Operations: There are two types of Operations on RDDs

1. Transformation: which create a new data-set from an existing one. For example, map is a transformation that passes each data-set element through a function and returns a new RDD representing the results
2. Action: return a value to the driver program after running a computation on the dataset. For example, reduce is an action that aggregates all the elements of the RDD using some function and returns the final result to the driver program In Scala, RDDs are created by an existing Scala collection in the driver program, and transforming it.

Spark Streaming Spark Streaming is an extension of the core Spark API that enables scalable, high-throughput, fault-tolerant stream processing of live data streams. This meets our need for stream processing of live twitter data stream. It also enables the

end user to process data much quicker than the time consumed when processing in a batch processing manner (like in Hadoop), thus saving time and money from a business perspective of the end user. Processed data can be pushed out to lesystems, databases, and live dashboards. For this Use-Case implementation we will push our processed twitter and Facebook data to HDFS and save it as an "external" Hive Table. Spark Streaming receives live input data streams and divides the data into batches, which are then processed by the Spark engine to generate the nal stream of results in batches. Spark Streaming provides a high-level abstraction called discretized stream or DStream, which represents a continuous stream of data. DStreams can be created either from input data streams from sources such as Kafka, Flume, and Kinesis, or by applying high-level operations on other DStreams. Internally, a DStream is represented as a sequence of RDDs. Any operation applied on a DStream translates to operations on the underlying RDDs. Integrating Input Flume DStream in Spark Streaming Spark Streaming provides two categories of built-in streaming sources.

Basic sources: Sources directly available in the Streaming Context API. Examples: le systems, socket connections, and Akka actors.

Advanced sources: Sources like Kafka,Flume,Kinesis,Twitter,etc. are available through extra utility classes. In Order to understand Flume Integration with Spark ,consider the following Schematic.

Push-based Receiver: Flume is designed to push data between Flume agents. Using this approach, Spark Streaming sets up a receiver that acts as an Avro sink for Flume. We congure Flume to push the data into the Avro sink. which incorporate Big Data Analytics in their use cases to derive/extract untapped business value e.g. greater/deeper insights into the minds of target customers.

STEP 1: CONFIGURING FLUME AGENT

STEP 2: CREATING A SCALA PROJECT TO HANDLING TWITTER INPUT STREAM

STEP 2.1: CREATE PROJECT DIRECTORY STRUCTURE

STEP 2.2: PROGRAMMING THE INITIALIZATION OF SPARK

1. **Create Spark Context:** The first thing a Spark program will do is to create a SparkContext object, which tells Spark how to access a cluster. To create a SparkContext I'll first need to build a SparkConf object that contains information about my this application.
2. **Initialize Streaming Context:** To initialize a Spark Streaming program, a StreamingContext object has to be created which is the main entry point of all Spark Streaming functionality.

3. **Creating an Input DStream Receiver Object using Flume Source:** As mentioned a *DStream discretized stream* is the abstraction provided to represent the stream of input data received from streaming sources. Every input DStream *except filestream* is associated with a Receiver object which receives the data from a source and stores it in Spark's memory for processing.

STEP 2.3: PROGRAMMING THE HANDLING OF INPUT TWITTER and FACEBOOK STREAM DATA:

STEP 2.3.1: Parsing the json object inside DStream:

In order to parse the json object and store information in form of scala class objects, we have chosen to:

1. Parse json object using JSONs, a library for parsing JSON objects in Scala.
2. Extract information from only the needed fields of the JSON object and store it as native Scala class object.

STEP 3: USING A SENTIMENT ANALYSIS ALGORITHM TO EVALUATE SENTIMENT

For sentiment analysis we are using the libraries which hit the end points of "hashtag" of twitter. Taking the data streams into account and putting them into RDDs and saving it's length into an object, the libraries then check the length of the sentiment to generate the weighted sentiment which is further converted into Sentiment Positive, Negative or neutral.

STEP 4: PERSISTING THE TWEET OBJECT IN A HIVE TABLE

STEP 4.1: CREATING A DATAFRAME FROM EXISTING RDDs AND SAVING AS HIVE TABLES

STEP 5: COMPILING THE PROJECT

STEP 5.1: Installing SBT In order to run sbt commands , we first need to install the tool in our host operating system, which in this case is Oracle Red Hat Linux:

STEP 5.2: CREATING THE BUILD DEFINITION FILE

STEP 5.3: CREATING A JAR FILE

STEP 6: DEPLOYMENT

Terminal 1: Navigate to the project root folder and execute the following command:
`spark-submit -class "main.scala.SampleApp" target/scala-2.10/sample-project2.11 - 1.0.jar`

Terminal 2: *Navigate to*

`~/usr/lib/flume-ng/bin/`

Directory path and run the following command : `./flume-ng-agent -nTwitterAgent - cconf - f/usr/lib/flume-ng/conf/flume.conf`

```

for (sentence <- annotation.get(classOf[CoreAnnotations.SentencesAnnotation])) {
  val tree = sentence.get(classOf[SentimentCoreAnnotations.AnnotatedTree])
  val sentiment = RNNCoreAnnotations.getPredictedClass(tree)
  val partText = sentence.toString

  if (partText.length() > longest) {
    mainSentiment = sentiment
    longest = partText.length()
  }
  sentiments += sentiment.toDouble
  sizes += partText.length
  println("debug: " + sentiment)
  println("size: " + partText.length)
}
val averageSentiment:Double = {
  if(sentiments.size > 0) sentiments.sum / sentiments.size
  else -1
}
val weightedSentiments = (sentiments, sizes).zipped.map((sentiment, size) => sentiment * size)
var weightedSentiment = weightedSentiments.sum / (sizes.fold(0)(_ + _))
if(sentiments.size == 0) {
  mainSentiment = -1
  weightedSentiment = -1
}
println("debug: main: " + mainSentiment)

```

Figure 5.25: Sentiment Analysis

Once deployed the Spark application can run on top of HDFS Cluster in YARN or Local Mode. The tweets along with the sentiment of each tweet can be viewed by querying the Hive table created, using a Hive Shell.

5.3.4 Big Data Discovery

Now after spark have performed sentiment analysis on the data i.e. tweets and posts. It will save the data into HIVE tables. Now I'll use Oracle Big Data Discovery to generate dashboards on those tables.

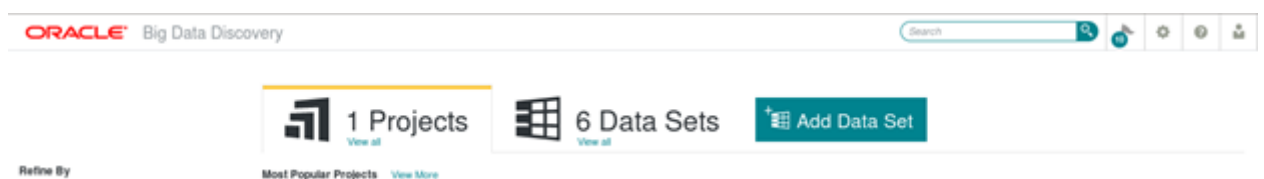


Figure 5.26: Big Data Discovery

Now I'll go to Oracle data integrator open the work repository

I will generate the Twitter and Facebook scenarios.

After generating scenarios, I'll make the ODI's packages for twitter and Facebook which will take the data form Hive table and will generate the Dashboards and define the overall workflow. Behind flume command the flume is updating the hashtag and downloading

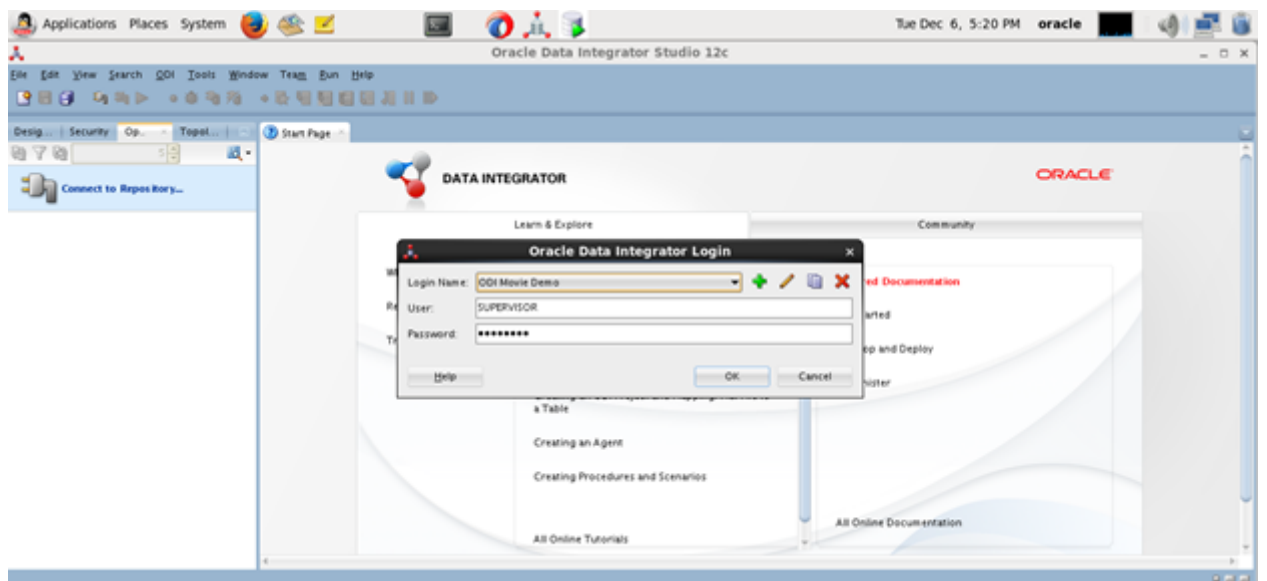


Figure 5.27: Oracle Data Integrator

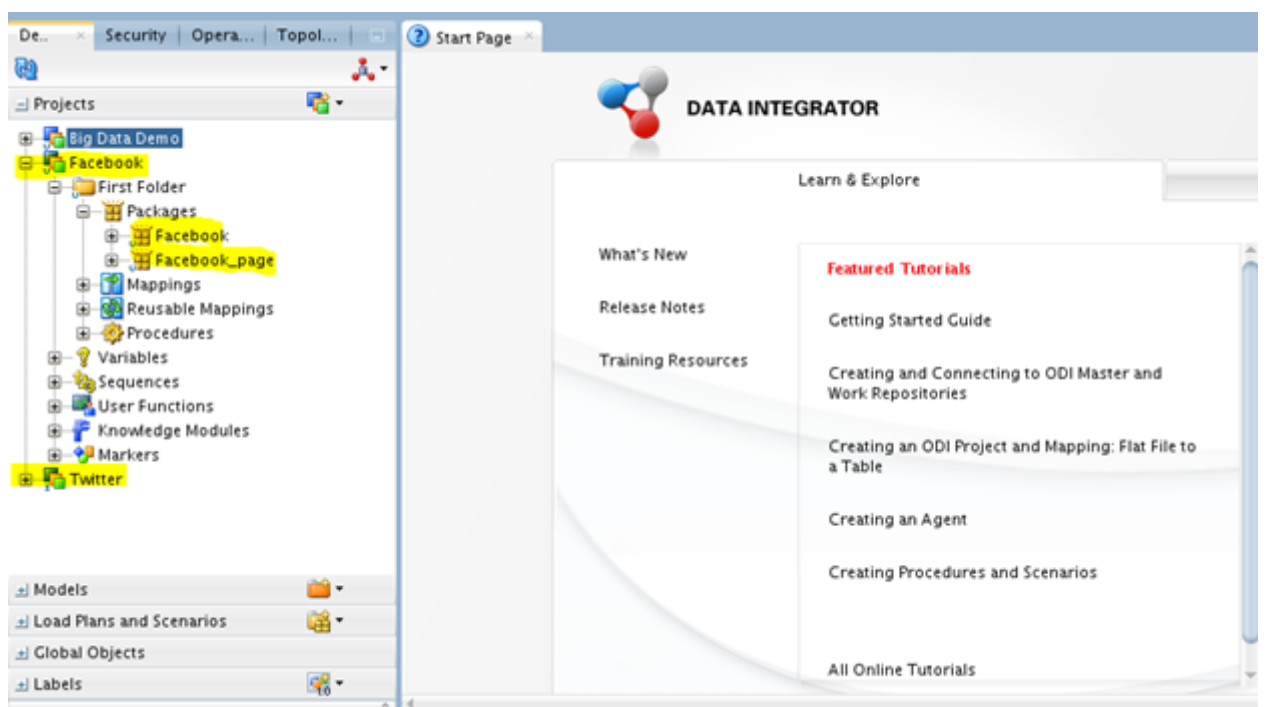


Figure 5.28: Generating Scenarios

the tweets Behind insert_into_hive the flume is throwing on the 44444 port and spark is listening. Behind sentiment Spark is performing the sentiment analysis Behind dashboards the whole scenarios is building up and generating the dashboards on run time tweets and posts.

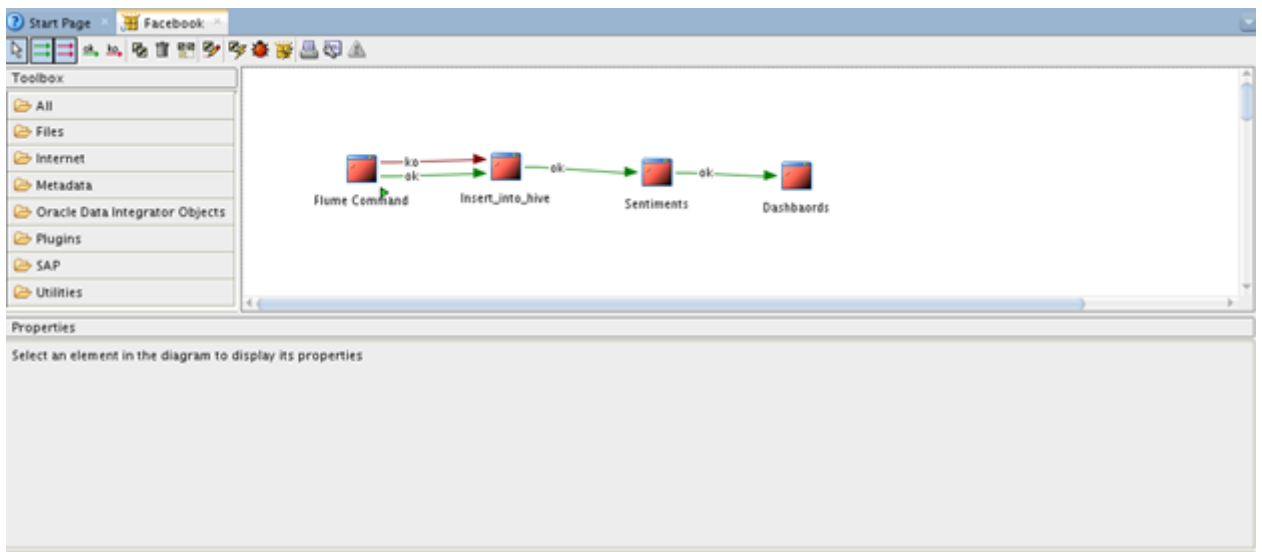


Figure 5.29: Facebook Scenario

Chapter 6

System Testing and Evaluation

Testing is included in each phase of project life cycle, yet it is different at every level of software development and has diverse destinations. In this section different programming testing strategies which have been utilized to assess the application have been depicted trailed by the test outcomes, finishing with a decision about the final item. Testing is included in each phase of project life cycle, yet it is different at every level of software development and has diverse destinations.

6.1 Software Testing Techniques

Software testing is a procedure by which we find the faults and errors in our system. We check our system whether it is fully filling the requirements of our project or not. We check all the functionalities to make sure our system is according to our gathered or required requirements. There are many testing software spectrum:

6.1.1 Unit Testing

Unit testing is done at the most minimal level. It tests the essential unit of programming, which is the smallest testable bit of programming, and is regularly called module. **Our system:** I will test whether the data is downloading from Twitter and Facebook or not. This is our one module.

6.1.2 Integration Testing

It is performed when at least two tried units are joined into a bigger structure. The test is regularly done on both the interfaces between the segments and the bigger structure being built, if its quality property can't be surveyed from its segments. **Our system:** I will test the connection between the virtual machine and HADOOP file system. Flume Integration with Spark. Integrate the whole project into a single jar file. Deploy and run the packaged application on the Spark Cluster integrating the whole together.

6.1.3 System Testing

I will test the whole system. The system will perform all the functional and nonfunctional requirements to see end to end quality of this system. **Our system:** I will test the system for all functional and nonfunctional requirements. Specifically I'll make sure no private data of any user is downloaded.

6.1.4 Structural Testing

Structural testing is viewed as White Box Testing. We test the system knowing it's whole structure and methodologies. In this testing all the emphasizing is on internal structure and data flow. **Our system:** I will test system's data flow. Whether the data is flowing in the right path and direction.

6.1.5 Performance Testing

In this testing we check the whole performance of the system. I'll check how accurate, how reliable the system is as it involved data of the users.

6.1.6 Stress Testing

Testing the points of confinement of the framework. Generally done by over-burdening the framework. The application was tried by checking different hashtags *incaseoftwitter*.

6.1.7 Configuration Testing

The testing will include testing all the configuration files. Whether they are compatible with each other or not. We will make sure we are using the right version of Virtual machine to gain the effects of HADOOP and spark is integrated with.

6.1.8 Security Testing

I'll make sure no private data for any user is downloaded and privacy of every user is held carefully. In case of Facebook the key provided by the admin of the group will be kept secret and will not be shared with any one. Will check the system by entering the wrong key and make sure it doesn't download anything.

6.1.9 Acceptance Testing

Testing all points of confinement of the framework. Generally done by over-burdening the framework. The application was tried by checking different hashtags (in case of twitter) and post in case of Facebook. Whether the system is acceptable with its full functionality or not.

6.2 Test Cases

6.2.1 Test Case # 1

Case ID	01
Name	Open the web page
Description	User will open the web page to start interaction with system.
Requirements	Internet connection and web page should exist.
Steps to be taken	I will access the web page on localhost of VM
Expected Result	The web page will be displayed
Actual Result	Web page is displayed
Status	Success
Remarks	N/A

Table 6.1: Test Case 01

6.2.2 Test Case # 2

Case ID	02
Name	Enter HASHTAG
Description	The user will open the Twitter tab and will enter the required hashtag
Requirements	Internet connection, web page and twitter tab should exist be accessible and display the box to enter HASHTAG
Steps to be taken	Click the Twitter tab and enter the required HASHTAG in the box
Expected Result	The tweets will start downloading
Actual Result	Starts downloading
Status	Success
Remarks	N/A

Table 6.2: Test Case 02

6.2.3 Test Case # 3

Case ID	03
Name	Run the script
Description	The user will open the Facebook tab and will enter the required name of the group.
Requirements	Internet connection, web page and Facebook tab should exist be accessible and display the box to enter the name of the group
Steps to be taken	Click the Facebook tab and enter the required HASHTAG in the box
Expected Result	The post will start downloading
Actual Result	Starts downloading
Status	Success
Remarks	N/A

Table 6.3: Test Case 03

6.2.4 Test Case # 4

Case ID	04
Name	Sentiment
Description	The user will click the sentiment analysis button to start performing it on the downloaded data.
Requirements	Internet connection, there should be file downloaded containing tweets or posts.
Steps to be taken	The spark will start streaming
Expected Result	The result showing whether the downloaded data was positive, negative or neutral
Actual Result	Starts downloading
Status	Success
Remarks	N/A

Table 6.4: Test Case 03

Chapter 7

Conclusions

Social Media analytics system was successfully implemented and it produced the effective and reliable results. This system was composed of a web page providing access to two social sites

1. Twitter
2. Facebook

The tweets containing the word we enter as hashtag starts downloading from twitter as soon as we enter the hashtag running the twitter 4j libraries and scripts at backend in the form of JSON. For the Facebook as soon as we enter the group name, the python script written for it starts downloading the group's post running python and shell script at backend. All this data will be on the fly (memory) and spark using spark stream and Scala will perform sentiment analysis on the downloaded tweets or post. Then the data after sentiment analysis will be sent to HIVE in the form of tables and then using the oracle BDD (Big data discovery) we will display the dashboards showing the result of the data whether it is positive, negative or neutral.

7.1 Limitations and Future Enhancement

For now we are downloading data from Facebook group that is private when it's key is provided, we can further expand it to all type of groups and public posts of users and pages. For twitter we can use the geographical API's as well to download the tweets which comes under certain diameter to your location. Social media analytics is limited to twitter and Facebook only, in future we can expand it to get data from google plus, instagram and snapchat.

Appendix A

User Manual

References

- [1] IBM. 4v's of big data. <http://www.ibmbigdatahub.com/tag/587>, 2016. Cited on p. 3.
- [2] Apache. Hadoop. <http://hadoop.apache.org/>, 2016. Cited on pp. 7 and 26.
- [3] Techopedia. Flat file database. <https://www.techopedia.com/definition/7231/flat-file-database-database>, 2016. Cited on p. 8.
- [4] Apache. Flume. <https://flume.apache.org/>, 2016. Cited on pp. 9 and 27.
- [5] Hortonworks. Spark. <http://hortonworks.com/apache/spark/>, 2016. Cited on pp. 9 and 27.
- [6] Apache. Spark streaming. <http://spark.apache.org/streaming/>, 2016. Cited on p. 9.
- [7] Oracle. Data integrator. <http://www.oracle.com/technetwork/middleware/data-integrator/overview/index.html>, 2016. Cited on p. 27.