

Statement by Candidate

I wish to state that the work embodied in this report titled “Media labs” is my own/the group contribution to the work carried out under guidance of Prof.Abhishek Mane at Veermata Jijabai Technological Institute. This work has not been submitted by any other Degree or Diploma of any University/Institute. Whenever references have been made to previous works of other, it has been clearly indicated.

Signature of Candidates:

Priyank Jain _____

Prateek Jakate _____

Jigar Bhati _____

Akash dhotre _____

CERTIFICATION

This is to certify that a student of B.Tech as a member of team of students has completed the project “Media Labs” to my satisfaction.

Prof. Abhishek Mane

Project Guide

Prof. Dr.B.B.Meshram

Head of Department

CERTIFICATION

This is to certify that a student of B.Tech as a member of team of students has completed the project “Media Labs” to my satisfaction.

Prof. Abhishek Mane

Project Guide

Examiner

ACKNOWLEDGEMENTS

We would like to thank our project guide Prof. Abhishek Mane for contributing his time and effort to help us with our project. His suggestions and guidelines have been of a great help during the entire course of our project. He has always been involved by discussing our project at each phase to make sure that the experiment is designed and carried out in an appropriate manner and that the our conclusions are appropriate, given their results. His constant support and interaction have been a driving force which has constantly motivated us to explore the different aspects of our project. We would also like to thank Dr B.B. Meshram, Head of the Computer Technology Department for his guidance and motivation.

Abstract

Media Labs is a social media analytics tool to measure the popularity of a TV show on Twitter. TRP ratings are not a reliable measure to understand the popularity of a TV show as TRP ratings take into account the taste of a very small group of audience as compared to the entire group of audience who view the TV show. Technologies to be used include: Twitter API, Java, MongoDB, Pig, Hadoop, MySQL, PHP, Javascript, HTML, CSS. A number of analytical results would be shown to the users to gain deep insights into the popularity of a TV show and a character. The dimensions along which analysis would take place include: Sentiment, Gender, Age group, Impressions: Unique/Repeated, Users: Unique/Repeated, Location, Device used.

Table Of Contents

Chapter 1: Introduction	9
1.1 Review	10
1.1.1 Target rating point	10
1.1.2 Gross rating Point	11
1.1.3 Reach	12
1.2 Television Indusrty In India	13
1.3 Audience metrics	14
1.3.1 DART	14
1.3.2 TAM & INTAM	14
1.3.3 aMap	15
1.4 Drawback of TRP rating	16
 Chapter 2 : Related work	 19
2.1 Background	20
2.2 Advantages of Media Labs over Topsy	21
 Chapter 3: Requirement Engineering	 23
3.1 Application Requirement	24
3.2 System Requirement	25
3.3 Survey Overview	26
 Chapter 4: Design	 29
4.1 use case diagram	30
4.2 System Architecture	31
4.3 Database Schema Diagram	32
4.4 EER Diagram	33
4.5 Component Diagram	34

4.6 Deployment Diagram	35
Chapter 5: Fetcher	36
5.1 Data Source: Twitter REST API	37
5.1.1 How Does Twitter REST API Work?	37
5.1.2 How to use the Twitter Search API?	37
5.1.3 How to build a query	40
5.2 Java Programming Language	42
5.3 Fetching tweets of a show using	
Twitter Search API and Twitter4J	42
5.3.1 Data Integration Step	45
5.3.1.1 Sentiment Analysis	46
5.3.1.2 Gender Analysis	47
5.3.1.3 Inserting into MongoDB	47
Chapter 6: Analyzer	49
6.1 Database – MongoDB	50
6.2 Database – MySQL	54
6.3 Software Framework: Hadoop	59
6.4 Programming Tool (High-Level Platform): Pig Latin	66
6.5 Database Connector: JDBC	70
6.6 Working of The Analyser	72
Chapter 7: User interface	75
7.1 HTML	76
7.2 CSS	77
7.3 JAVACRIPT	81
7.4 JQUERY	82
7.5 PHP	85

7.6 BOOTSTRAP	87
7.7 GOOGLE CHARTS	90
7.8 Implementation	92
Chapter 8: Future work and Conclusion	96
8.1 Future work	97
8.2 Conclusion	98
Chapter 9: References	99

Chapter 1

Introduction

1.1 Review

Television in India is a huge industry which has thousands of programs in many languages. The small screen has produced numerous celebrities, some even attaining national fame. More than half of all Indian households own a television. As of 2012, the country has a collection of free and subscription services over a variety of distribution media, like the CHERIAN channel, through which there are over 823 channels of which 184 are pay channels.

As the number of people watching television has increased rapidly, it has become a broad market for the various advertising firms. These firms are now a days more focussed towards broadcasting their ads on television. For this purpose they have to know which ad to be broadcasted on what time and on which channel. For this purpose the concept of TRP has been introduced.

1.1.1 Target Rating Points (TRP)

Target Rating Points (TRPs) are the gross rating points delivered by a media vehicle to a specific target audience.

Purpose of TRP

The purpose of the 'target rating point' metric is to measure impressions in relation to the number of people in a specific target audience for an advertisement. Thus, the TRP is a measure of the purchased points representing an estimate of the component of the target audience within the gross audience. Similar to the gross rating point, it is measured as the sum of ratings achieved by a specific media vehicle (e.g., TV channel or program) of the target audience reached.

Construction

Target rating points (TRPs) quantify the gross rated points achieved by an advertisement or campaign among targeted individuals within a larger population.

For example, if an advertisement appears more than once, the entire gross audience also, the TRP figure is the sum of each individual GRP, multiplied by the estimated target audience in the gross audience. The TRP and GRP metrics are both critical components for determining the marketing effectiveness of a particular advertisement. Outside of television, TRPs are calculated using the denominator of the total target audience, and the numerator as the total impressions delivered to this audience x 100. (As in $1,000,000$ impressions among the target audience / $10,000,000$ people in total in the target audience x 100 = 10 TRPs). TRPs are often added up by week, and presented in a flowchart so a marketer can see the amount of impressions delivered to the target audience from each media channel.

1.1.2 Gross rating point

Gross rating point (GRP) is a term used in advertising to measure the size of an audience reached by a specific media or schedule. Specifically, GRPs quantify impressions as a percentage of the population reached rather than in absolute numbers reached. Target rating points express the same concept, but with regard to a more narrowly defined target audience.

GRPs are used predominantly as a measure of media with high *potential* exposures or impressions.

Purpose

The purpose of the GRP metric is to measure impressions in relation to the number of people in the audience for an advertising campaign. GRP values are commonly used by [media buyers](#) to compare the advertising strength of various media vehicles.

Construction

GRPs are the product of the percentage of the audience **reached** by an advertisement, times the **frequency** they see it in a given **campaign** (frequency × % reached).^[4]

$$\text{GRPs (\%)} = \text{Reach (\%)} \times \text{Average frequency (\#)}$$

A **television advertisement** that is aired five times reaching 50% of the audience each time it is aired would have a GRP value of 250 (5 × 50%). To achieve a common denominator and compare media, (reach × frequency) are expressed over time (divided by time) to determine the 'weight' of a media campaign.

Alternatively, GRPs may be calculated in relation to the number of impressions:

$$\text{GRPs (\%)} = 100 * \text{Impressions (\#)} \div \text{Defined population (\#)}$$

1.1.3 Reach (advertising)

In the application of statistics to advertising and media analysis, reach refers to the total number of different people or households exposed, at least once, to a medium during a given period. Reach should not be confused with the number of people who will actually be exposed to and consume the advertising, though. It is just the number of people who are exposed to the medium and therefore have an opportunity to see or hear the ad or commercial. Reach may be stated either as an absolute number, or as a fraction of a given population (for instance 'TV households', 'men' or 'those aged 25–35').

For any given viewer, they have been "reached" by the work if they have viewed it at all (or a specified amount) during the specified period. Multiple viewings by a single member of the audience in the cited period do not increase reach; however, media people use the term effective reach to describe the quality of exposure. Effective reach and reach are two different measurements for a target audience who receive a given message or ad.

Since reach is a time-dependent summary of aggregate audience behavior, reach figures are meaningless without a period associated with them: an example of a valid reach figure would be to state that "[example website] had a one-day reach of 1565 per million on 21 March 2004" (though unique users, an equivalent measure, would be a more typical metric for a website).

Reach of television channels is often expressed in the form of "x minute weekly reach" - that is, the number (or percentage) of viewers who watched the channel for at least x minutes in a given week.

1.2 Television industry in India

Television in India is a huge industry which has thousands of programmes in many languages. As per the TAM Annual Universe Update - 2010, India now has over 134 million households (out of 223 million) with television sets, of which over 103 million have access to Cable TV or Satellite TV, including 20 million households which are DTH subscribers. In Urban India, 85% of households have a TV and over 70% of all households have access to Satellite, Cable or DTH services. TV owning households have been growing at between 8-10%, while growth in Satellite/Cable homes exceeded 15% and DTH subscribers grew 28% over 2009. (However, some analysts place the number of households with television access at closer to 180 million since roughly a third of all rural families may watch television at a neighbouring relatives home, and argue that Cable TV households are probably closer to 120 million owing to a certain percentage of informal/unregistered Cable Networks that aren't counted by mainstream surveys). It is also estimated that India now has over 823 TV channels covering all the main languages spoken in the nation.

2013 list of Top 10 television shows

Rank	Series	Genre	Network	Air date	Air time	Avg. viewership (millions)	Peak viewership (millions)

1	<i>Yeh Rishta Kya Kehlata Hai</i>	Soap opera	STAR Plus	All year	Mon-Sat 9:30PM IST	8.8	9.7
2	<i>Taarak Mehta Ka Ooltah Chashmah</i>	Sitcom	SAB TV	All year	Mon-Fri 8:30PM IST	8.4	9.3
3	<i>Diya Aur Baati Hum</i>	Soap opera	Colors	All year	Mon-Sat 8:00PM IST	6.8	8.9
4	<i>Pyaar Ka Dard Hai</i>	Soap opera	STAR Plus	All year	Mon-Sat 10:00PM IST	6.7	7.9
5	<i>Jodha Akbar</i>	Historical drama	Zee TV	Since June 18, 2013	Mon-Fri 8:00PM IST	6.6	6.6
6	<i>Comedy Nights with Kapil</i>	Comedy/Talk show	Colors	Since June 22, 2013	Sat-Sun 10:00PM IST	6.6	8.6
7	<i>Saath Nibhaana Saathiya</i>	Soap opera	STAR Plus	All year	Mon-Sat 7:00PM IST	6.4	8.7
8	<i>Nach Baliye 5</i>	Reality/Dance	STAR Plus	December 29, 2012 – March 23, 2013	Sat-Sun 9:00PM IST	6.0	7.1

9	<i>Qubool Hai</i>	Soap opera	Zee TV	All year	Mon-Fri 9:30PM IST	6.0	7.4
10	<i>Mahabharat</i>	Mythological drama	STAR Plus	Since September 16, 2013	Mon-Sat 8:30PM IST	5.6	7.2

1.3 Audience metrics

Television metrics in India have gone through several phases in which it fragmented, consolidated and then fragmented again.

1.3.1 DART

During the days of the single channel Doordarshan monopoly, DART (Doordarshan Audience Research Team) was the only metric available. This used the notebook method of recordkeeping across 33 cities across India. DART continues to provide this information independent of the Private agencies. DART till this date is the only rating system that still measures audience metrics in Rural India.

1.3.2 TAM & INTAM

In 1994, claiming a heterogeneous and fragmenting television market ORG-MARG introduced INTAM (Indian National Television Audience Measurement). Ex-officials of DD (Doordarshan) claimed that INTAM was introduced by vested commercial interests who only sought to break the monopoly of DD and that INTAM was significantly weaker in both sample size, rigour and the range of cities and regions covered.

In 1997, a joint industry body appointed TAM (backed by AC Nielsen as the official recordkeeper of audience metrics. Due to the differences in methodology and samples of TAM and INTAM, both provided differing results for the same programs.

In 2001, a confidential list of households in Mumbai that were participating in the monitoring survey was released, calling into question the reliability of the data. This subsequently led to the merger of the two measurement systems into TAM. For several years after this, in spite of misgivings about the process, sample and other parameters, TAM was the defacto standard and monopoly in the audience metrics game.

1.3.3 aMap

In 2004, a rival ratings service funded by American NRI investors, called Audience Measurement Analytics Limited (aMap) was launched. Although initially, it faced a cautious uptake from clients, the TAM monopoly was broken.

What differentiates aMap is that its ratings are available within one day as compared to TAM's timeline of one week.

1.4 Drawback of TRP rating

Currently the only electronic rating agency operating in our country is INTAM (Indian Television Audience Management). The two methodologies it adopts for determining TRP's are

1. Frequency Monitoring: In this method, meters are installed in the homes which are considered as the sample of the total population. These meters keep track of programmes/channel being surfed by the particular household. It reads the frequencies of channels, which are later, decoded into the name of the channels and the agency prepares a national data on the basis of its sample homes readings. But there is a drawback in the technique, as

cable operators frequently change the frequencies of the different channels before sending signals to the homes

2. Picture Matching: It is a relatively new concept to India and in this technique the meters tracks data of the picture that being watched in the household. Along with this agency also records all the channels' data in the form of small picture portion. Data collected from the sample homes is later on matched with the main data bank to interpret the channel name

Though this method has been accepted is the TRP system foolproof. The answer is NO. The major reason why it can't be considered foolproof is due to the selection of the sample. The ratings derived are measured based on the data collected from top sixteen cities of nine states in India. This sample further does not take into consideration the people from lower middle class and also people from smaller towns. Further they have not even considered the elite class on the assumption that these people would prefer to watch high end English channels as compared to the local channels.

The most crucial part of TRP business is the sample size of the research. Presently, TRP is based upon only a small urban sample of 5500 homes spread all over India. Most of the sample homes are situated in urban areas. Critics doubt as to how this small sample could truly represent the taste of Indian. That's why Doordarshan has its own ratings system **DART (Doordarshan Audience Ratings)**. DART is a **diary based system of ratings**. DD people distribute diaries in sample homes and the viewers are asked to note down each programme as and when watched by family members. In the end of the week a person collects all the diaries and sends them to the head office, where popularity of programmes is calculated.

So that brings us to a conclusion that TRP's just provide a glimpse of a small sample and not the actual mood of the population. Certain sections of the society have not been represented in this rating and hence it does not give the correct reflection of the viewing habits. So there is a need to have a rating based on the viewing habits of elite, middle and lower class spread across all the cities of our country for this rating to be meaningful for the channels and advertisers. But it also true that true viewership habits can never be

estimated as it is not possible to reach every single home in this country. So this method does provide a hazy picture which nevertheless has been appreciated by the industry. Advertisers, channels, producers would get a somewhat skewed picture of their standing in the fiercely competitive television market.

Chapter 2

Related work

2.1 Background

In this chapter we will discuss the related works, which contain twitter analysis . Later in the experiments result section, we will make some comparisons between our method and the related works.

Topsy is a social search and analytics company based in San Francisco, California. The company is a certified Twitter partner and maintains a comprehensive index of tweets, numbering in hundreds of billions, dating back to Twitter's inception in 2006. Topsy makes products to search, analyze and draw insights from conversations and trends on the public social websites including Twitter and Google+

Social Indices

Twitter Political Index

This index was co-developed by Twitter and Topsy. It debuted in August 2012 and originally compared social sentiment for the two primary American presidential candidates.

Twitter Oscars Index

This index was also co-developed by Twitter and Topsy. It debuted in January 2013 and originally compared social sentiment for films nominated for Academy awards in six categories: Best Picture, Best Actor, Best Actress, Best Supporting Actor, Best Supporting Actress and Best Director. Topsy sentiment analysis used in this index correctly predicted five out of the six award recipients.

SXSW Trendspotter

In March 2013, Mashable and Topsy co-produced the Mashable SXSW Trendspotter, which is a mobile-enabled website that visitors see what's trending at the SXSW event, based on real-time analysis of Twitter conversations. The SXSW Trendspotter provides analysis of:

- Which topics are trending
- Which start-ups, brands, bands and films are getting generating popular social conversations
- Which SXSW sessions and events are the most popular
- Details about the top tweets, news, photos and videos around each topic

2.2 Advantages of media labs over Topsy

Gender analysis

Gender analysis is a type of socio-economic analysis that uncovers how gender relations affect a development problem. The aim may just be to show that gender relations will probably affect the solution, or to show how they will affect the solution and what could be done. Gender analysis frameworks provide a step-by-step methodology for conducting gender analysis

Mainly in gender analysis we are going to consider the ratio of male and female who are tweeting on twitter. The analysis will describe about the no of male and female who show there concern to some tv shows.

Gender wise sentiments

Along with normal gender analysis, gender wise sentimental analysis is also provided by media labs. The number of positive, negative and neutral tweets made by males and females are shown in this case. This analysis will give a clear view of the audience in India.

Device analysis

In case of device analysis the device are shown on which the tweet were performed. The device where the tweets are tweeted could be mobile or laptop in case of media labs. It give the analytical result and also the number of tweet from laptop or mobile.

Impression analysis

An impression (in the context of online advertising) is a measure of the number of times a tweet is retweeted and favorited, each time a tweet is retweeted or favorite it is counted as one impression.

For each show and character, number of retweets and favorites of all the tweets corresponding to it are displayed as Impression analysis. This explains the intensity of a show and the popularity of it.

Chapter 3

Requirement Engineering

3.1 Application Requirement

No	Requirement description	nature
1	The total tweets for every show and character	Must have
2	The number of positive, negative and neutral tweets for every show and character	Must have
3	The number of tweets for each one character for males and females	Must have
4	The number of positive, negative and neutral tweets for tv character by males	Must have
5	The number of positive, negative and neutral tweets for tv character by females	Must have
6	The number of tweets for each shows for males and females	Must have
7	The number of positive, negative and neutral tweets for tv show by males	Must have
8	The number of positive, negative and neutral tweets for tv show character by female	Must have
9	The number of tweets made from mobile for shows	Nice to have
10	The number of tweets made from mobile for character	Nice to have
11	The number of tweets made from pc for shows	Nice to have
12	The number of tweets made from pc for character	Nice to have
13	The number of retweet for tv show and character	Nice to have
14	The number of favorite for tv show and character	Nice to have

3.2 System Requirement

No	Requirement Description
1	The application can fetch around 180 query in the time interval of 15 minutes
2	The twitter api can provide some limited query for a single search and that is 100 query
3	The operator which are used by the application to fetch the tweets should not be greater than 10
4	The tweet that are fetched from the twitter api can be only for 7 days
5	Both the fetcher and analyzer must be started and must run together.
6	Fetcher should give up automatically if the threshold is acquired
7	Program should be able to handle network exception rate limit , and no tweet exception
8	The analyzer application must wait until the tweets for an hour are retrieved, and then start performing the analysis.
9	The analyzer application at a time summarizes only the tweets inserted in one hour and must group the tweets created in the same hour.
10	The analyzer must for each show and character must count the tweets based on various parameters
11	After inserting the summary, the analyzer application must update the latest time of summary.

3.3 Survey Overview

As a part of the requirement gathering process, we conducted a small online survey.

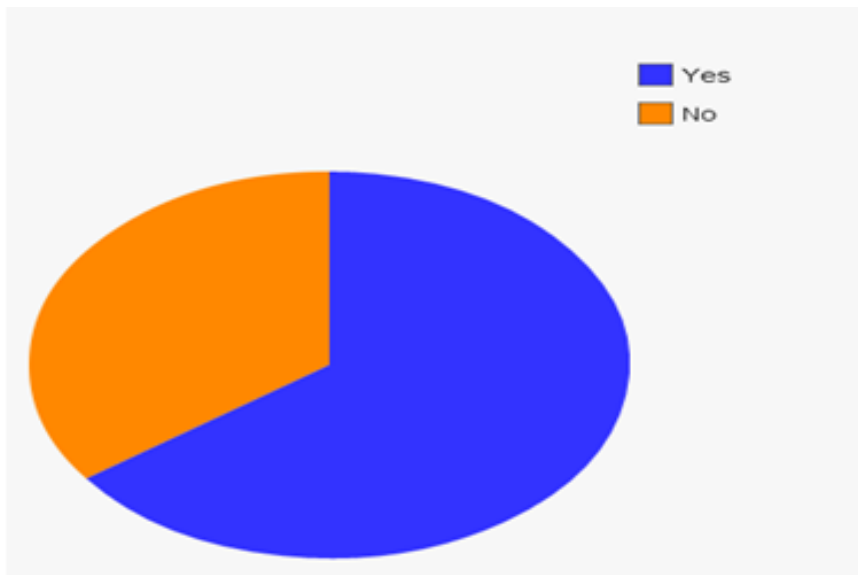
The survey was intended to gather information regarding the following aspects:

- ❑ Will the application be used?
- ❑ What should be the platform used?
- ❑ Are the people in India tweeting on tv shows?
- ❑ What do the targeted audience think about TRP?

We floated the survey in the targeted audience and asked them the some questions.

Some important questions and their answers are summarized in the next section.

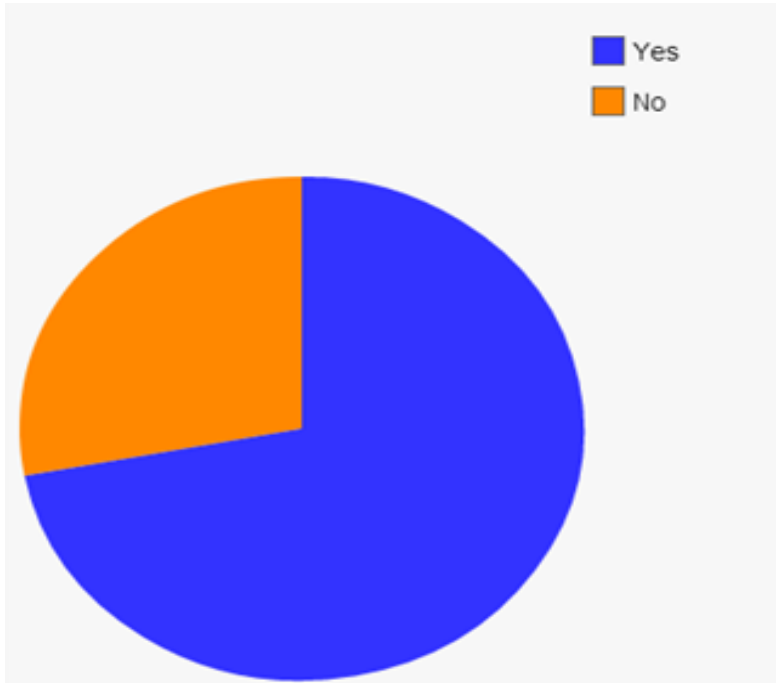
Q1 Do you have a twitter account?



65% people say yes

35% people say no

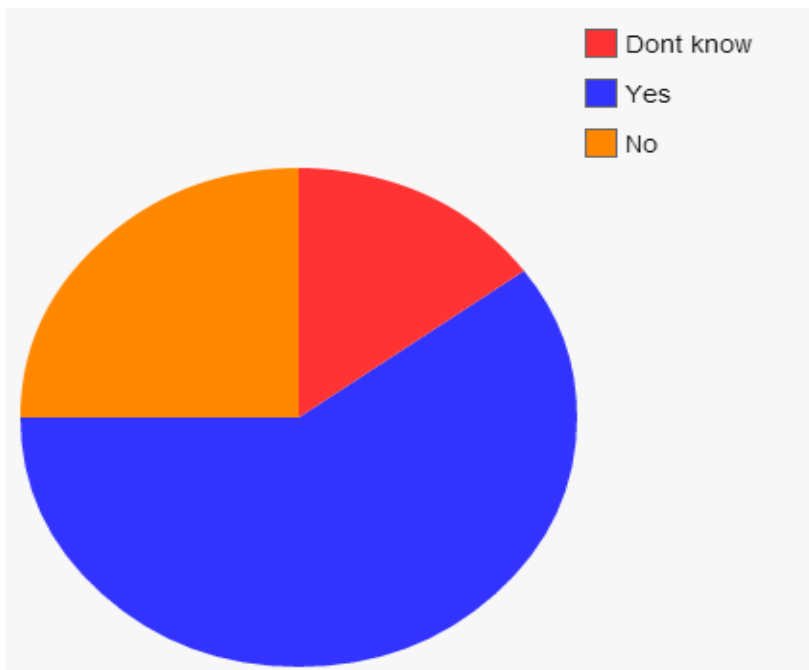
Q2 Do you tweet on tv shows and character?



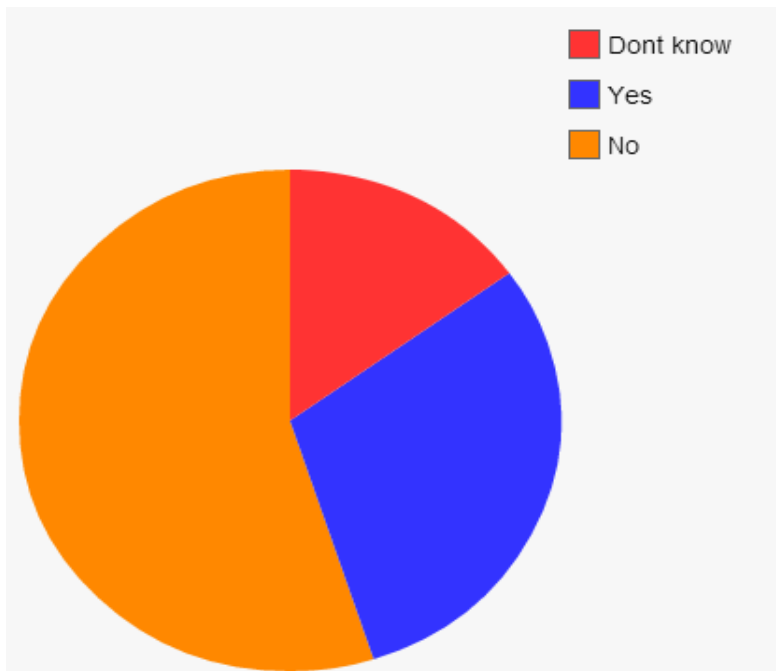
70% people say yes

30% people say no

Q3 Do you know what TRP means?



Q4 Do you agree that TRP gives correct rating?



55% says no

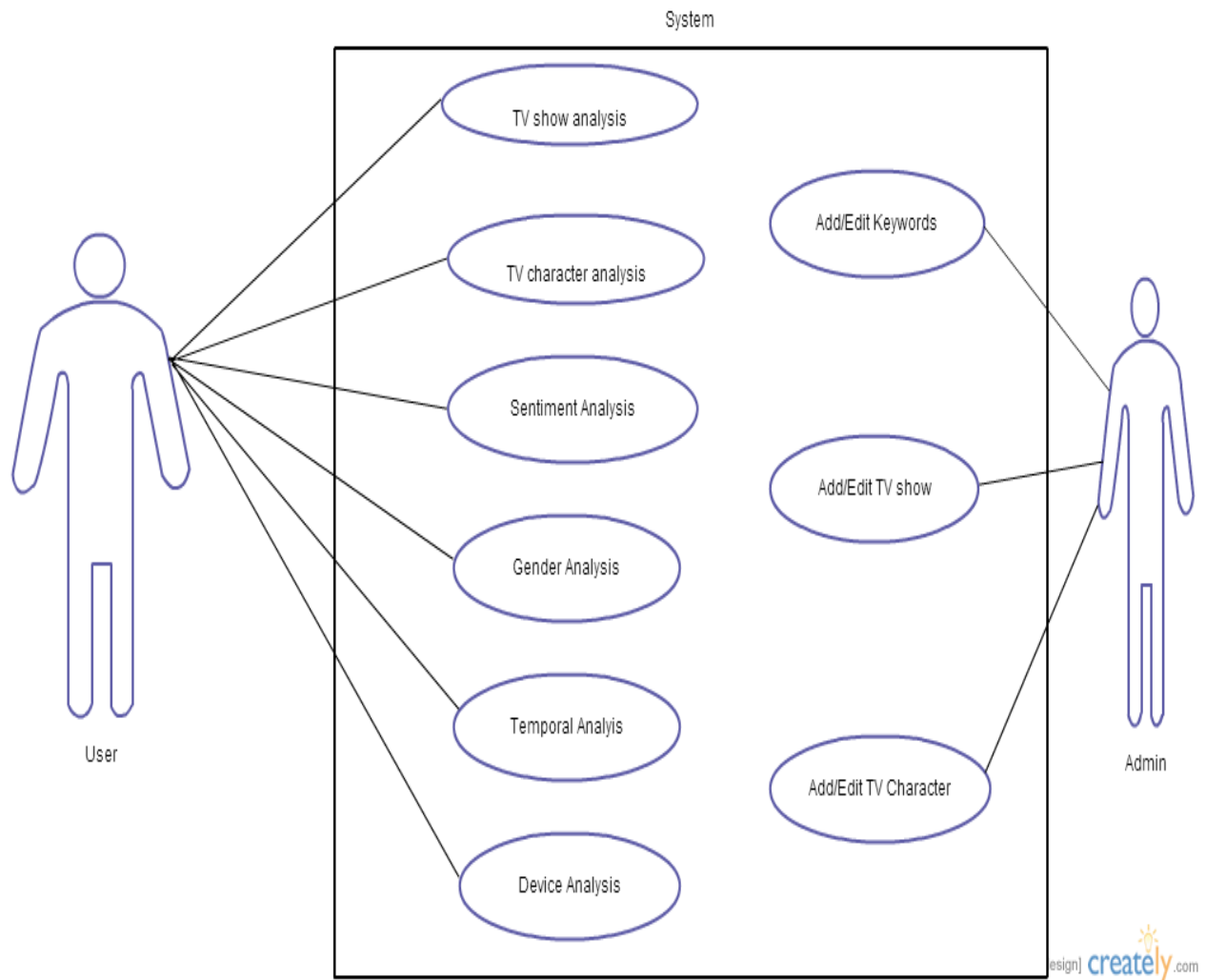
30% says yes

15% says don't know

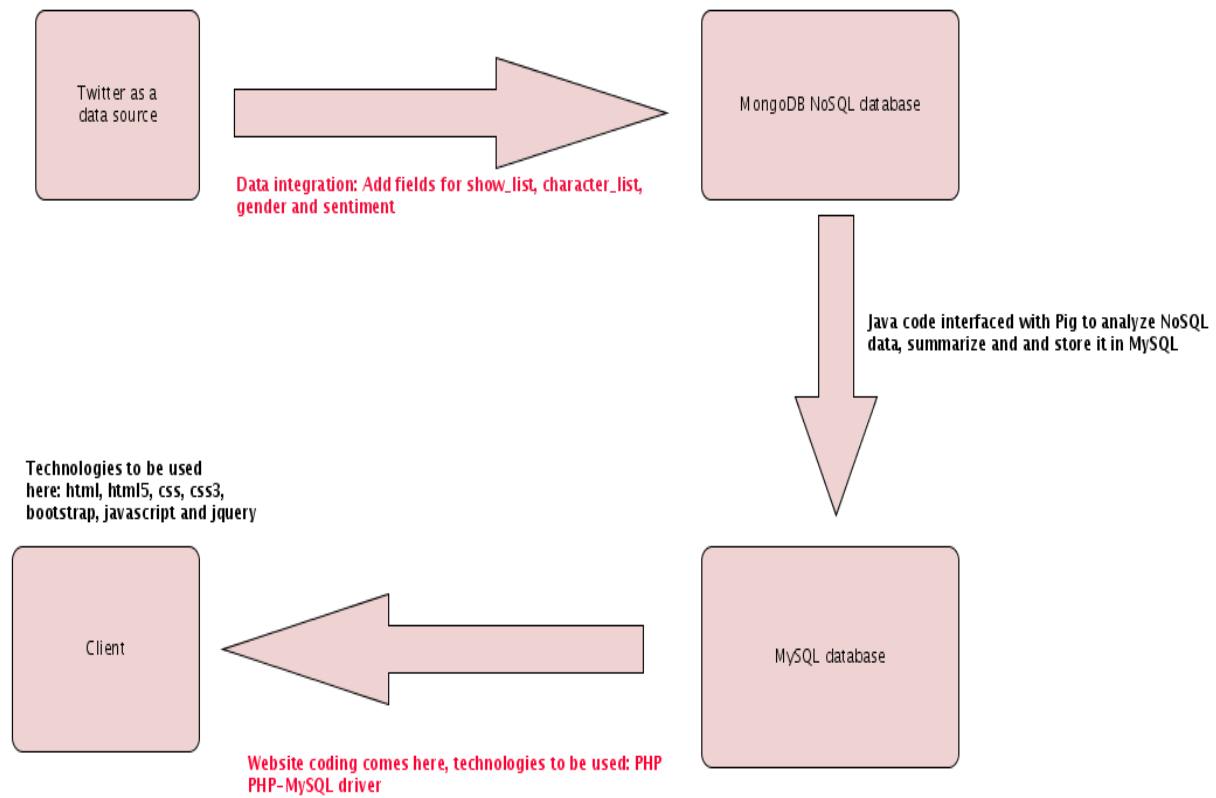
Chapter 4

Design

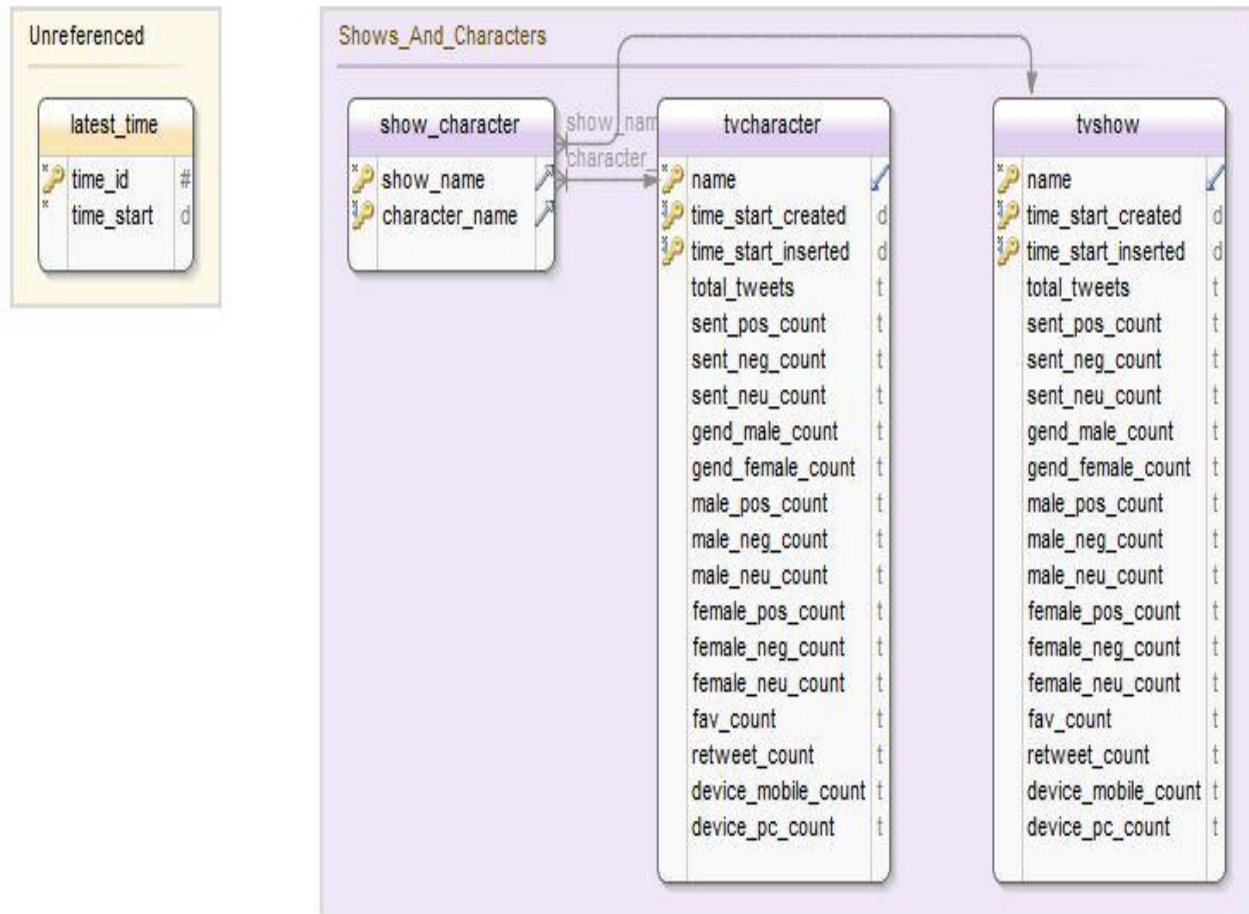
4.1 use case diagram



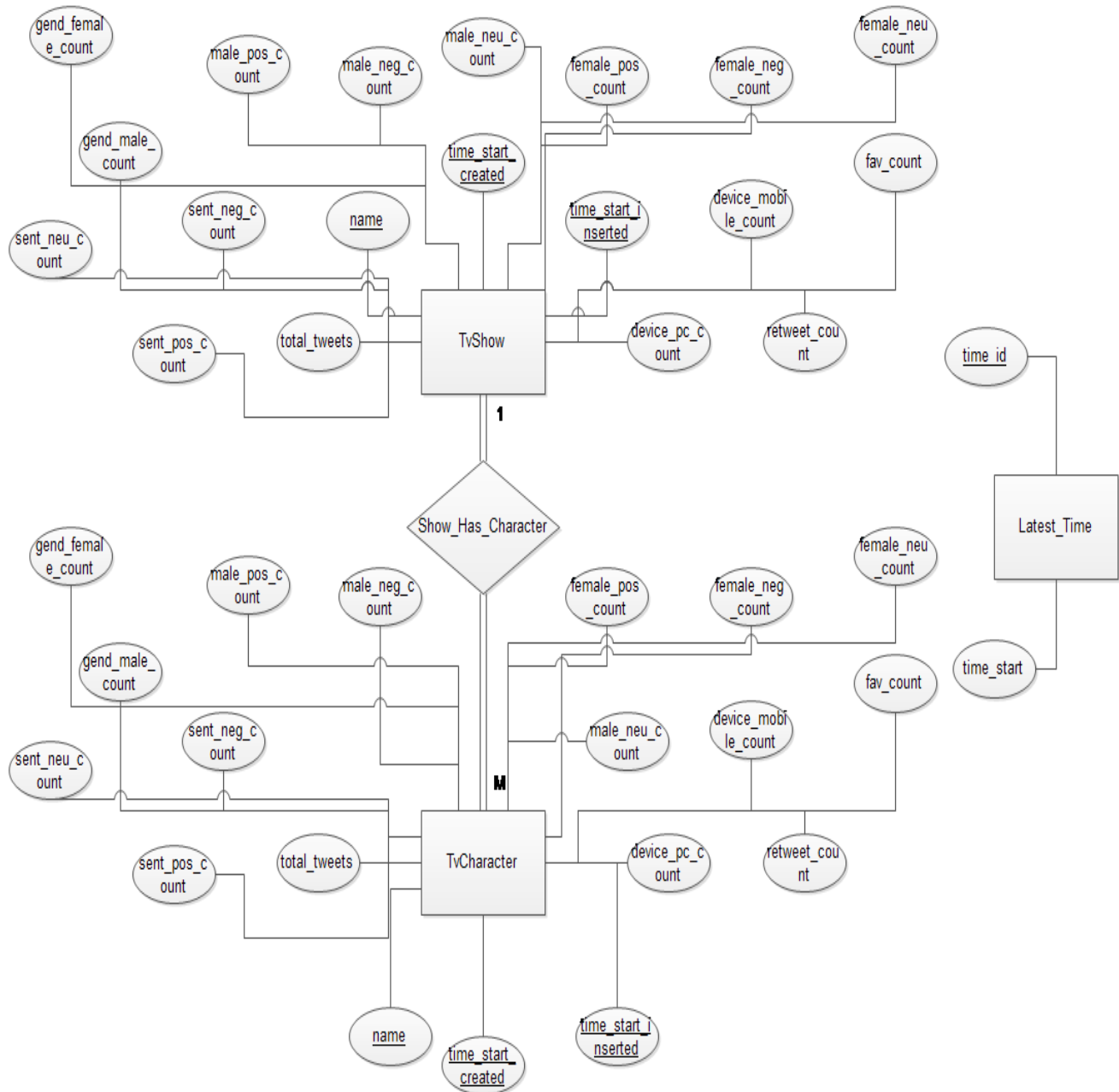
4.2 System Architecture



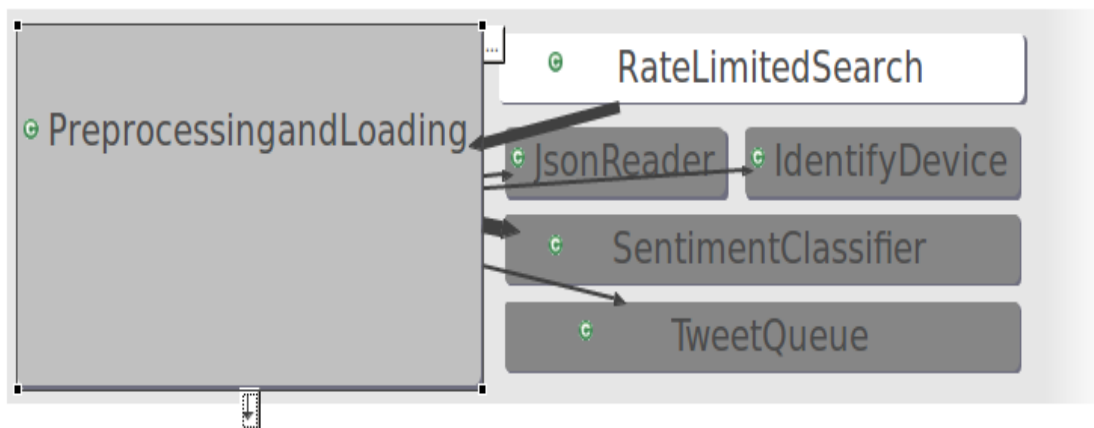
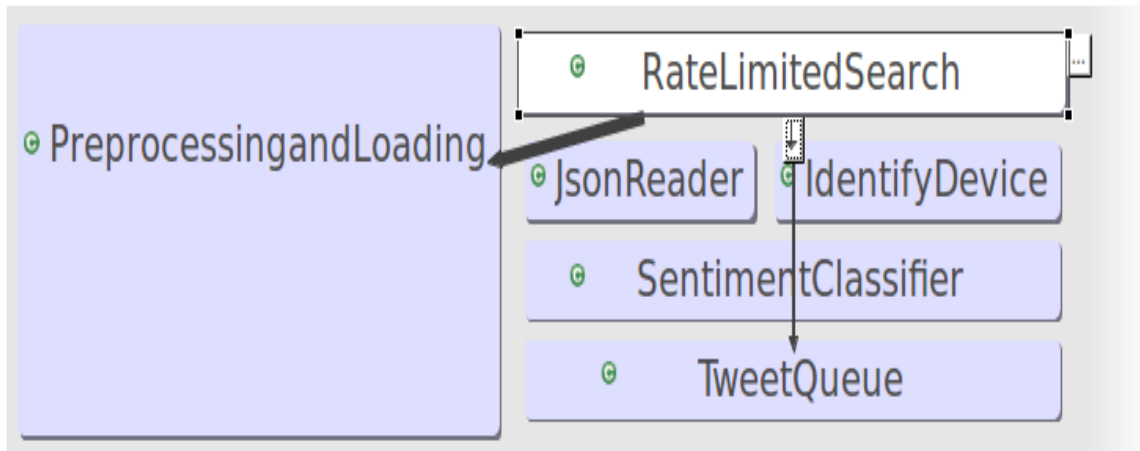
4.3 Database Schema Diagram



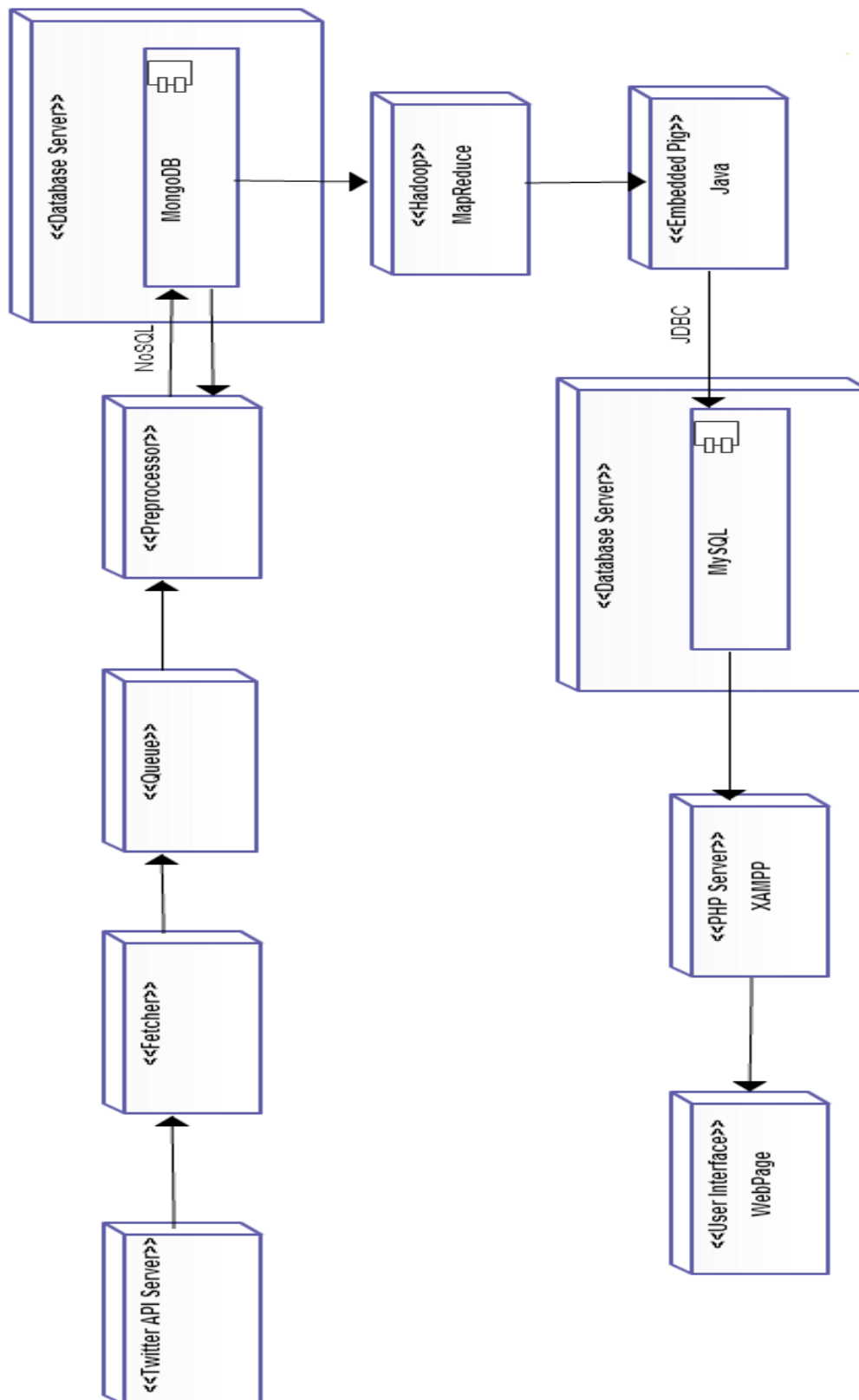
4.4 EER Diagram



4.5 Component Diagram



4.6 Deployment Diagram



Chapter 5

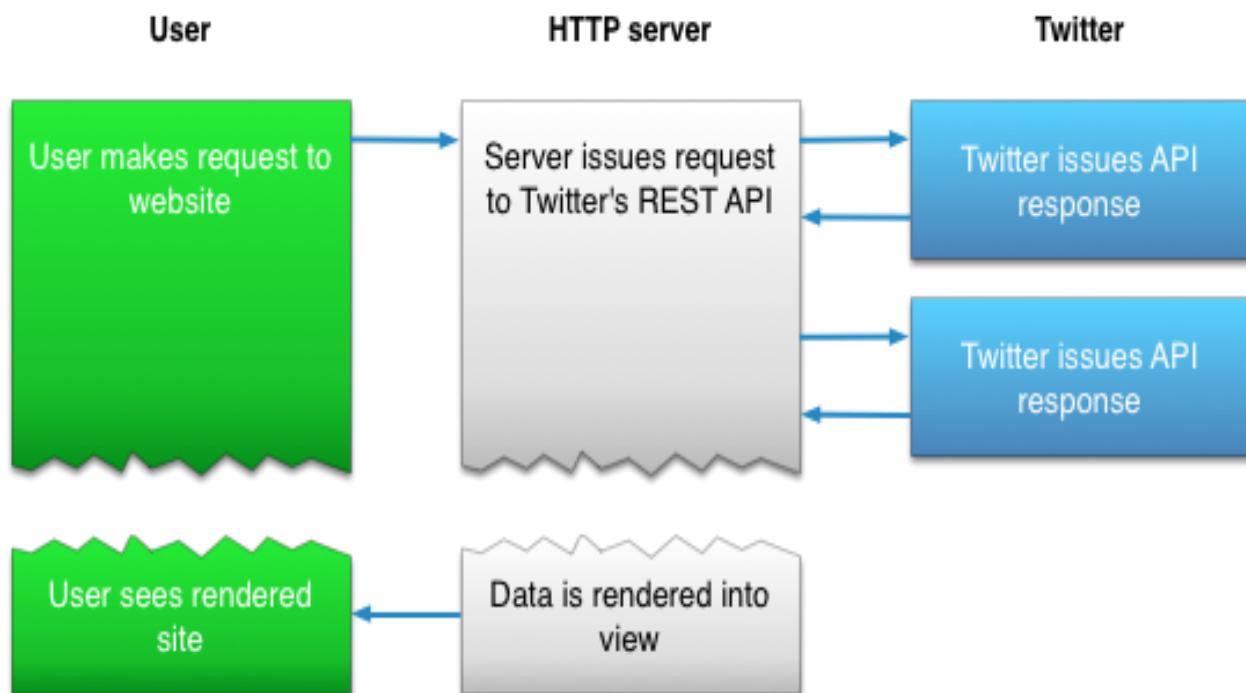
Fetcher

5.1 Data Source: Twitter REST API

We are specifically using the Search module in the Twitter REST API. List below are the details of the same:

The Twitter Search API returns a collection of Tweets matching a specified query. Twitter's search service and, by extension, the Search API is not meant to be an exhaustive source of Tweets. Not all Tweets will be indexed or made available via the search interface.

5.1.1 How Does Twitter REST API Work?



5.1.2 How to use the Twitter Search API?

Resource URL

<http://search.twitter.com/search.format>

q required

A UTF-8, URL-encoded search query of 1,000 characters maximum, including operators. Queries may additionally be limited by complexity.

Example Values: @noradio

geocode optional

Returns tweets by users located within a given radius of the given latitude/longitude. The location is preferentially taking from the Geotagging API, but will fall back to their Twitter profile. The parameter value is specified by "latitude,longitude,radius", where radius units must be specified as either "mi" (miles) or "km" (kilometers). Note that you cannot use the near operator via the API to geocode arbitrary locations; however you can use this geocodeparameter to search near geocodes directly. A maximum of 1,000 distinct "sub-regions" will be considered when using the radius modifier.

Example Values: 37.781157,-122.398720,1mi

lang optional

Restricts tweets to the given language, given by an ISO 639-1 code. Language detection is best-effort.

Example Values: eu

locale optional

Specify the language of the query you are sending (only ja is currently effective). This is intended for language-specific consumers and the default should work in the majority of cases.

Example Values: ja

result_type optional

Optional. Specifies what type of search results you would prefer to receive. The current default is "mixed." Valid values include:

- * mixed: Include both popular and real time results in the response.

- * recent: return only the most recent results in the

response

* popular: return only the most popular results in the response.

Example Values: mixed, recent, popular

count optional

The number of tweets to return per page, up to a maximum of 100. Defaults to 15. This was formerly the "rpp" parameter in the old Search API.

Example Values: 100

until optional

Returns tweets generated before the given date. Date should be formatted as YYYY-MM-DD. Keep in mind that the search index may not go back as far as the date you specify here.

Example Values: 2012-09-01

since_id optional

Returns results with an ID greater than (that is, more recent than) the specified ID. There are limits to the number of Tweets which can be accessed through the API. If the limit of Tweets has occurred since the since_id, the since_id will be forced to the oldest ID available.

Example Values: 12345

max_id optional

Returns results with an ID less than (that is, older than) or equal to the specified ID.

Example Values: 54321

include_entities optional

The entities node will be disincluded when set to false.

Example Values: false

callback optional

If supplied, the response will use the JSONP format with a callback of the given name. The usefulness of this parameter is somewhat diminished by the requirement of authentication for requests to this endpoint.

Example Values: processTweets

5.1.3 How to build a query

The best way to build a query and test if it's valid and will return matched Tweets is to first try it at twitter.com/search. As you get a satisfactory result set, the URL loaded in the browser will contain the proper query syntax that can be reused in the API endpoint. Here's an example:

1. We want to search for tweets referencing @twitterapi account. First, we run the search on twitter.com/search
2. Check and copy the URL loaded. In this case, we got: <https://twitter.com/search?q=%40twitterapi>
3. Replace "<https://twitter.com/search>" with <https://api.twitter.com/1.1/search/tweets.json> and you will get: <https://api.twitter.com/1.1/search/tweets.json?q=%40twitterapi>
4. Execute this URL to do the search in the API

Query operators

The query can have operators that modify its behavior, the available operators are:

Operator	Finds tweets...
watching now	containing both "watching" and "now". This is the default operator.
"happy hour"	containing the exact phrase "happy hour".

Operator	Finds tweets...
love OR hate	containing either "love" or "hate" (or both).
beer -root	containing "beer" but not "root".
#haiku	containing the hashtag "haiku".
from:alexiskold	sent from person "alexiskold".
to:techcrunch	sent to person "techcrunch".
@mashable	referencing person "mashable".
superhero since:2010-12-27	containing "superhero" and sent since date "2010-12-27" (year-month-day).
ftw until:2010-12-27	containing "ftw" and sent before the date "2010-12-27".
movie -scary :)	containing "movie", but not "scary", and with a positive attitude.
flight :(containing "flight" and with a negative attitude.
traffic ?	containing "traffic" and asking a question.
hilarious filter:links	containing "hilarious" and linking to URL.
news source:twitterfeed	containing "news" and entered via TwitterFeed

5.2 Java Programming Language

Java Programming Language was used for the fetcher because of the availability of the strong Twitter4J library.

Twitter4J is an unofficial Java library for the Twitter API.

Twitter4J is featuring:

- ✓ 100% Pure Java - works on any Java Platform version 5 or later
- ✓ Android platform and Google App Engine ready
- ✓ Zero dependency : No additional jars required
- ✓ Built-in OAuth support
- ✓ Out-of-the-box gzip support
- ✓ 100% Twitter API 1.1 compatible

System Requirements

OS: Windows or any flavor of Unix that supports Java.

JVM: Java 5 or later

How To Use

Just add twitter4j-core-4.0.1.jar to your application classpath.

5.3 Fetching tweets of a show using Twitter Search API and Twitter4J

To begin with the fetching of tweets related to a particular TV show, we need to have the various keywords for the TV show and it's characters. The keywords have to be gathered

through a deep research about the TV show and it's characters on Twitter and by actually watching a few episodes of the TV shows. The keywords are then organized as per TV show's characters or keywords of character's inside an XML file which is then read by the fetcher JAVA code. Here is sample of how the shows.xml file looks like

Shows.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<shows>

    <show>

        <name>Comedy Nights</name>

        <characters>

            <character>

<title>Bittu</title>

<keywords>@KapilSharmaK9,KapilSharma,#kapil,bittu</keywords>

                <since>0</since>

            </character>

        </characters>

    </show>

</shows>
```

Note that a keyword may contain:

1. Twitter Handle(s) (starting with # symbol)
2. Hashtag(s) (starting with @ symbol)
3. Terms with one or more words

Now that we have the keywords to be searched for, we move onto the process of sending request to Twitter Search API for these keywords.

Here is a sample code snippet demonstrating the use of Twitter4J library for the search API

```
package twitter4j.examples.search;

import twitter4j.*;

import java.util.List;

public class SearchTweets {
    public static void main(String[] args) {
        if (args.length < 1) {
            System.out.println("java twitter4j.examples.search.SearchTweets [query]");
            System.exit(-1);
        }
        Twitter twitter = new TwitterFactory().getInstance();
        try {
            Query query = new Query(args[0]);
            QueryResult result;
            do {
                result = twitter.search(query);
                List<Status> tweets = result.getTweets();
                for (Status tweet : tweets) {
                    System.out.println("@ " + tweet.getUser().getScreenName() + " - " +
tweet.getText());
                }
            } while ((query = result.nextQuery()) != null);
            System.exit(0);
        } catch (TwitterException te) {
```

```
        te.printStackTrace();  
        System.out.println("Failed to search tweets: " + te.getMessage());  
        System.exit(-1);  
    }  
}  
}
```

The program loops the search module for each TV show and each TV character, and this whole process goes on indefinitely. The returned data is then processed to construct the show_list and character_list for the tweet. The show_list for a tweet consists of comma separated names of TV shows to which a TV show is related to, the character list is similar, except that it corresponds to characters.

After each iteration of the Search module, the <since> tag for the particular TV show/character for which show is being carried out is then updated, so as to next retrieve those elements

5.3.1 Data Integration Step:

The JSON data retrieved from Twitter Search API is appended with show_list, character_list. The tweet is then entered into a ConcurrentQueue which is share across another thread ProcessProcessingAndLoading.java. The code inside RateLimitedSearch.java acts as the producer into the queue and PreProcessingAndLoading.java acts as the consumer from the other end of the queue.

The PreProcessing step inserts few more key-value pairs in the JSON object.

These includes gender information and sentiment information.

5.3.1.1 Sentiment Analysis:

For sentiment analysis we have used the LingPipe library, which is based on neural networks. The LingPipe takes a training dataset as an input and then uses those to make decisions whether a tweet is by a male or a female.

The SentimentClassifier.java file for sentiment analysis is shown below:

```
import com.aliasi.classify.LMClassifier;
import com.aliasi.util.AbstractExternalizable;
import com.aliasi.classify.ConditionalClassification;
import java.io.*;

public class SentimentClassifier {

    String[] categories;
    LMClassifier classifier;

    public SentimentClassifier() {

        try {
            classifier= (LMClassifier) AbstractExternalizable.readObject(new File("classifier.txt"));
            categories = classifier.categories();
        }
        catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
        catch (IOException e) {
            e.printStackTrace();
        }

    }
}
```

```
public String classify(String text) {  
    ConditionalClassification classification = classifier.classify(text);  
    return classification.bestCategory();  
}  
}
```

5.3.1.2 Gender Analysis:

For gender analysis, we are using the genderize.io API.

The API has a database of distinct names across various countries and languages.

At the moment, the database contains 86710 distinct names across 74 countries and 81 languages.

All requests are sent to the following base URL using GET.

GET <http://api.genderize.io>

Single Usage:

An example of genderizing a single name could look like this

GET <http://api.genderize.io?name=peter>

This would render a JSON response like the following. The count represents the number of data entries examined in order to calculate the response.

```
{"name":"peter","gender":"male","probability":"0.99","count":796}
```

5.3.1.3 Inserting into MongoDB:

Now that all the required data from different sources has been integrated into a single unified JSON, it is ready to be inserted into MongoDB NoSQL database.

We use the MongoDB Java driver available on the official MongoDB website. The relevant code to insert the unified Tweet JSON into MongoDB has been show below: import java.net.UnknownHostException;

```
public void insertIntoMongo(String tweet,String user,String tweet_id,String
user_id)throws UnknownHostException
{
    DB db=mongo.getDB("twitter");
    DBCollection tweetCollection=db.getCollection("tweets");
    BasicDBObject tweetquery=new BasicDBObject("id_str",tweet_id);
    if(tweetCollection.find(tweetquery).count()==0)
    {
        DBObject tweetObject=(DBObject)JSON.parse(tweet);
        tweetCollection.insert(tweetObject);
    }
    DBCollection userCollection=db.getCollection("users");
    BasicDBObject query=new BasicDBObject("id_str",user_id);
    if(userCollection.find(query).count()==0)
    {
        DBObject userObject=(DBObject)JSON.parse(user);
        userCollection.insert(userObject);
    }
}
```

The above code ensures that duplicate tweets or users are not re-inserted.

Chapter 6

Analyzer

The following tools will be used during the development of the Analyzer:

Platform: Linux (Ubuntu)

Programming language: Java, J2SE

IDE: Eclipse Kepler 4.3

Database: MongoDB, MySQL

Software Framework: Hadoop

Programming Tool (High-Level Platform): Pig Latin

Database Connector: JDBC

6.1 Database – MongoDB

MongoDB (from "humongous") is a cross-platform document-oriented database system. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas (MongoDB calls the format BSON), making the integration of data in certain types of applications easier and faster. Released under a combination of the GNU Affero General Public License and the Apache License, MongoDB is free and open-source software.

MongoDB is an open-source document database, and the leading NoSQL database. Written in C++, MongoDB features:

1) Document-Oriented Storage »

JSON-style documents with dynamic schemas offer simplicity and power.

2) Full Index Support »

Index on any attribute, just like you're used to.

3) Replication & High Availability »

Mirror across LANs and WANs for scale and peace of mind.

4) Auto-Sharding »

Scale horizontally without compromising functionality.

5) Querying »

Rich, document-based queries.

6) Fast In-Place Updates »

Atomic modifiers for contention-free performance.

7) Map/Reduce »

Flexible aggregation and data processing.

8) GridFS »

Store files of any size without complicating your stack.

9) MongoDB Management Service »

Monitoring and backup designed for MongoDB.

10) Professional Support by MongoDB »

Enterprise class support, training, and consulting available.

Reason for using MongoDB:

-Schema less: MongoDB is document database in which one collection holds different different documents. Number of fields, content and size of the document can be differ from one document to another.

-Structure of a single object is clear

-No complex joins

- Deep query-ability. MongoDB supports dynamic queries on documents using a document-based query language that's nearly as powerful as SQL
- Tuning
- Ease of scale-out: MongoDB is easy to scale
- Conversion / mapping of application objects to database objects not needed
- Uses internal memory for storing the (windowed) working set, enabling faster access of data

Use:

MongoDB is used to store information about the tweets and the users who have made those tweets.

Setup:

Run the following command in terminal:

```
sudo apt-get update
```

```
sudo apt-get install mongodb
```

Implementation:

Create a Database “twitter” containing the following:

Collections “tweets”, “users”.

“tweets” collection stores information about tweets and contains a reference user_id to “users” collection.

“users” collection stores information about the users who made the tweets.

View the contents:

```

hduser@ubuntu:~$ mongo
MongoDB shell version: 2.0.4
connecting to: test
> use twitter;
switched to db twitter
> db.tweets.find().sort( { _id : -1 } ).limit(1);
{ "_id" : ObjectId("534a71de8f42ae448d4cd2a8"), "retweeted" : false, "in_reply_to_screen_name" : null, "character_list" : "bittu", "possibly_sensitive" :
false, "truncated" : false, "lang" : "en", "in_reply_to_status_id_str" : null, "inserted_at" : "2014/04/13 16:45:42", "sentiment" : "neu", "show_list" :
"comedy nights", "retweet_id" : "null", "in_reply_to_user_id_str" : null, "in_reply_to_status_id" : null, "created_at" : "2014/04/13 15:23:29", "user_id"
: "1039185624", "favorite_count" : 0, "place" : null, "coordinates" : null, "metadata" : { "result_type" : "recent", "iso_language_code" : "en" }, "text"
: "#youtube Watch @kapilsharmaK9 does a 'Comedy Nights' with Kejriwal @ArvindKejriwal at IOTY #Aap http://t.co/hr2Dhl0puj #AAPpositive", "contributors" :
null, "geo" : null, "keywords" : "comedy nights", "entities" : { "symbols" : [ ], "urls" : [ { "expanded_url" : "http://www.youtube.com/watch?v=J
aAP-57ugeE&sns=tw", "indices" : [ 96, 118 ], "display_url" : "youtube.com/watch?v=JaAP-5...", "url" : "http://t.co/hr2Dhl0puj" } ], "hashtags" :
[ { "text" : "youtube", "indices" : [ 0, 8 ] }, { "text" : "Aap", "indices" : [ 91, 95 ] }, { "text" : "AAPpositive", "indices" : [ 119, 131 ] } ] }, "user_mentions" : [ { "id" : 1492538024, "name" : "kapil ", "indices" : [ 15
, 29 ], "screen_name" : "KapilSharmaK9", "id_str" : "1492538024" }, { "id" : 405427035, "name" : "Arvind Kejriwal", "i
ndices" : [ 67, 82 ], "screen_name" : "ArvindKejriwal", "id_str" : "405427035" } ] }, "source" : "<a href=\"http://twitter.com/tweetbutton
\" rel=\"nofollow\">Tweet Button</a>", "favorited" : false, "device" : "pc", "retweet_count" : 0, "in_reply_to_user_id" : null, "id_str" : "455282619421
167616" }
> db.users.find().sort( { _id : -1 } ).limit(1);
{ "_id" : ObjectId("534a71d88f42ae448a4cd2a8"), "location" : "mumbai", "default_profile" : true, "profile_background_tile" : false, "statuses_count" : 816
, "lang" : "en", "profile_link_color" : "0084B4", "inserted_at" : "2014/04/13 16:45:36", "profile_banner_url" : "https://pbs.twimg.com/profile_banners/232
9587008/1397134502", "following" : false, "protected" : false, "favourites_count" : 20, "profile_text_color" : "333333", "description" : "", "verified" :
false, "contributors_enabled" : false, "profile_sidebar_border_color" : "C0DEED", "name" : "ashutosh kumar dubey", "profile_background_color" : "C0DEED",
"gender" : "male", "created_at" : "Thu Feb 06 00:58:42 +0000 2014", "is_translation_enabled" : false, "default_profile_image" : false, "followers_count" :
7, "profile_image_url_https" : "https://pbs.twimg.com/profile_images/453880330257047552/6p-wC14y_normal.jpeg", "geo_enabled" : false, "profile_background
_image_url" : "http://abs.twimg.com/images/themes/theme1/bg.png", "profile_background_image_url_https" : "https://abs.twimg.com/images/themes/theme1/bg.pn
g", "follow_request_sent" : false, "entities" : { "description" : { "urls" : [ ] } }, "url" : null, "utc_offset" : null, "time_zone" : null, "notification
s" : false, "profile_use_background_image" : true, "friends_count" : 143, "profile_sidebar_fill_color" : "DDEEF6", "screen_name" : "ashutosh614", "id_str"
: "2329587008", "profile_image_url" : "http://pbs.twimg.com/profile_images/453880330257047552/6p-wC14y_normal.jpeg", "listed_count" : 0, "is_translator"
: false }
> db.tweets.count();
7055
> db.users.count();
3730
> exit
bye
hduser@ubuntu:~$

```

6.2 Database – MySQL

MySQL ("My Sequel") is the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius's daughter, My. The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

Reasons for using MySQL:

- 1) Relational Database System: Like almost all other database systems on the market, MySQL is a relational database system.
- 2) Client/Server Architecture: MySQL is a client/server system. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they query data, save changes, etc. The clients can run on the same computer as the server or on another computer (communication via a local network or the Internet).
- 3) SQL compatibility: MySQL supports as its database language -- as its name suggests -- SQL (Structured Query Language). SQL is a standardized language for querying and updating data and for the administration of a database.
- 4) User interface: There are a number of convenient user interfaces for administering a MySQL server.
- 5) Full-text search: Full-text search simplifies and accelerates the search for words that are located within a text field. If you employ MySQL for storing text (such as in an Internet discussion group), you can use full-text search to implement simply an efficient search function.

- 6) Foreign key constraints: These are rules that ensure that there are no cross references in linked tables that lead to nowhere. MySQL supports foreign key constraints for InnoDB tables.
- 7) Programming languages: There are quite a number of APIs (application programming interfaces) and libraries for the development of MySQL applications. For client programming you can use, among others, the languages C, C++, Java, Perl, PHP, Python, and Tcl.
- 8) Platform independence: It is not only client applications that run under a variety of operating systems; MySQL itself (that is, the server) can be executed under a number of operating systems. The most important are Apple Macintosh OS X, Linux, Microsoft Windows, and the countless Unix variants, such as AIX, BSDI, FreeBSD, HP-UX, OpenBSD, Net BSD, SGI Iris, and Sun Solaris.
- 9) Speed: MySQL is considered a very fast database program. This speed has been backed up by a large number of benchmark tests.

Use:

Used to store the first level summary that is inserted into it by embedded pig in java program using JDBC.

Setup:

Run the following commands in terminal:

```
sudo apt-get install mysql-server
```

```
sudo apt-get install mysql-client
```

During the installation process you will be prompted to enter a password for the MySQL root user.

Once the installation is complete, the MySQL server should be started automatically.

Implementation:

Run the following commands to create the database:

```
create database twitter;
use twitter;
create table tvshow(
name varchar(100),
time_start_created timestamp,
time_start_inserted timestamp,
total_tweets long,
sent_pos_count long,
sent_neg_count long,
sent_neu_count long,
gend_male_count long,
gend_female_count long,
male_pos_count long,
male_neg_count long,
male_neu_count long,
female_pos_count long,
female_neg_count long,
female_neu_count long,
fav_count long,
retweet_count long,
device_mobile_count long,
device_pc_count long,
primary key(name, time_start_created, time_start_inserted));

create table tvcharacter(
name varchar(100),
time_start_created timestamp,
```



```
time_start_inserted timestamp,  
total_tweets long,  
sent_pos_count long,  
sent_neg_count long,  
sent_neu_count long,  
gend_male_count long,  
gend_female_count long,  
male_pos_count long,  
male_neg_count long,  
male_neu_count long,  
female_pos_count long,  
female_neg_count long,  
female_neu_count long,  
fav_count long,  
retweet_count long,  
device_mobile_count long,  
device_pc_count long,  
primary key(name, time_start_created, time_start_inserted));
```

```
create table show_character(  
show_name varchar(100),  
character_name varchar(100),  
primary key(show_name, character_name));
```

```
create table latest_time(  
time_id int,  
time_start timestamp,  
primary key(time_id));
```

View the contents

```

@ubuntu: ~
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use twitter;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from tvshow where time_start_created='2014/04/13 16:00:00';
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| name      | time_start_created | time_start_inserted | total_tweets | sent_pos_count | sent_neg_count | sent_neu_count | gend_male_count | gend_fem | |
| ale_count | male_pos_count | male_neg_count | male_neu_count | female_pos_count | female_neg_count | female_neu_count | fav_count | retweet_count | device |
| _mobile_count | device_pc_count |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| comedy nights | 2014-04-13 16:00:00 | 2014-04-13 19:00:00 | 35          | 8              | 2              | 25             | 12          | 4          |
| 2            | 2              | 8              | 2          | 0              | 2              | 6              | 159         | 8          |
| 27           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select * from tvcharacter where time_start_created='2014/04/13 16:00:00';
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| name | time_start_created | time_start_inserted | total_tweets | sent_pos_count | sent_neg_count | sent_neu_count | gend_male_count | gend_female_coun | |
| t | male_pos_count | male_neg_count | male_neu_count | female_pos_count | female_neg_count | female_neu_count | fav_count | retweet_count | device_mobile_ |
| count | device_pc_count |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| bittu | 2014-04-13 16:00:00 | 2014-04-13 19:00:00 | 10          | 2              | 0              | 8              | 2            | 1            |
| 1      | 0              | 1              | 1          | 0              | 0              | 0              | 0            | 0            |
| 10     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.03 sec)

mysql> select * from latest_time;
+-----+-----+
| time_id | time_start      |
+-----+-----+
| 1       | 2014-04-13 19:00:00 |
+-----+-----+
1 row in set (0.00 sec)

mysql> exit

```

6.3 Software Framework: Hadoop

The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The project includes these modules:

- Hadoop Common: The common utilities that support the other Hadoop modules.
- Hadoop Distributed File System (HDFS™): A distributed file system that provides high-throughput access to application data.
- Hadoop YARN: A framework for job scheduling and cluster resource management.
- Hadoop MapReduce: A YARN-based system for parallel processing of large data sets.

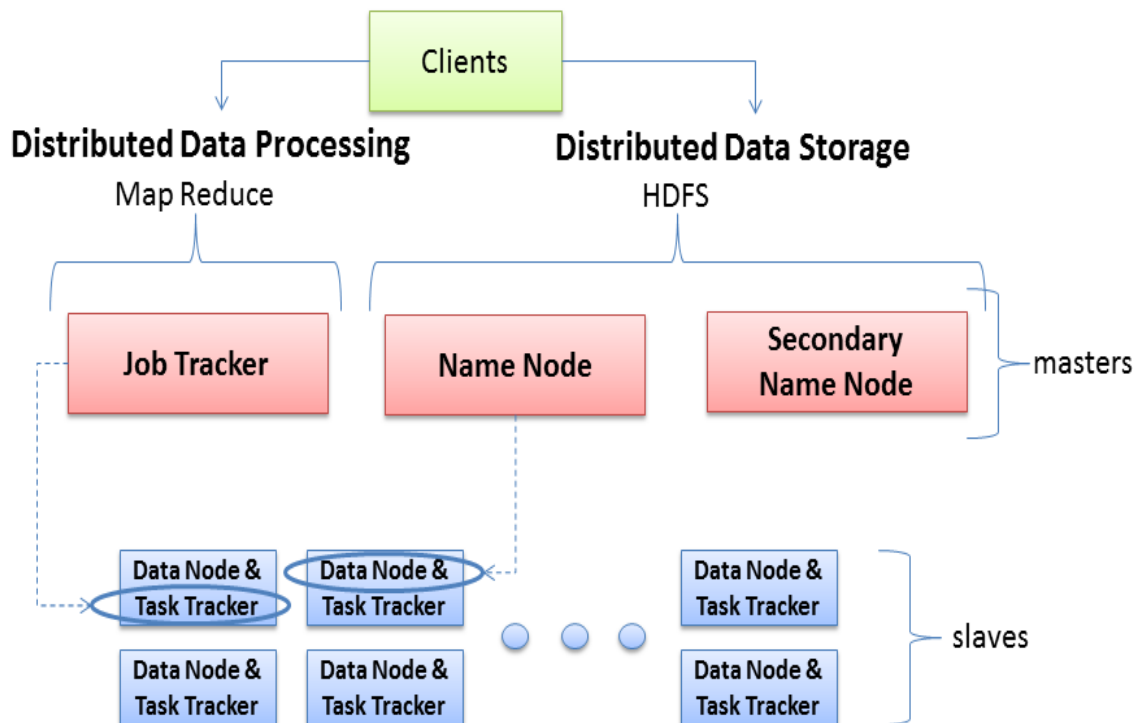
Hadoop distributed file system

The Hadoop distributed file system (HDFS) is a distributed, scalable, and portable file-system written in Java for the Hadoop framework. Each node in a Hadoop instance typically has a single namenode; a cluster of datanodes form the HDFS cluster. The situation is typical because each node does not require a datanode to be present. Each datanode serves up blocks of data over the network using a block protocol specific to HDFS. The file system uses the TCP/IP layer for communication. Clients use Remote procedure call (RPC) to communicate between each other.

JobTracker and TaskTracker: the MapReduce engine

Above the file systems comes the MapReduce engine, which consists of one JobTracker, to which client applications submit MapReduce jobs. The JobTracker pushes work out to available TaskTrackernodes in the cluster, striving to keep the work as close to the data as possible. With a rack-aware file system, the JobTracker knows which node contains the data, and which other machines are nearby. If the work cannot be hosted on the actual node where the data resides, priority is given to nodes in the same rack. This reduces network traffic on the main backbone network. If a TaskTracker fails or times out, that part of the job is rescheduled. The TaskTracker on each node spawns off a separate Java Virtual Machine process to prevent the TaskTracker itself from failing if the running job crashes the JVM. A heartbeat is sent from the TaskTracker to the JobTracker every few minutes to check its status. The Job Tracker and TaskTracker status and information is exposed by Jetty and can be viewed from a web browser.

Hadoop Server Roles



Reason for using Hadoop:

1) Scalable:

Hadoop is a highly scalable storage platform, because it can store and distribute very large data sets across hundreds of inexpensive servers that operate in parallel. Unlike traditional relational database systems (RDBMS) that can't scale to process large amounts of data, Hadoop enables businesses to run applications on thousands of nodes involving thousands of terabytes of data.

2) Cost effective:

Hadoop also offers a cost effective storage solution for businesses' exploding data sets. The problem with traditional relational database management systems is that it is extremely cost prohibitive to scale to such a degree in order to process such massive volumes of data. In an effort to reduce costs, many companies in the past would have had to down-sample data and classify it based on certain assumptions as to which data was the most valuable.

3) Flexible:

Hadoop enables businesses to easily access new data sources and tap into different types of data (both structured and unstructured) to generate value from that data. This means businesses can use Hadoop to derive valuable business insights from data sources such as social media, email conversations or clickstream data. In addition, Hadoop can be used for a wide variety of purposes, such as log processing, recommendation systems, data warehousing, market campaign analysis and fraud detection.

4) Fast:

Hadoop's unique storage method is based on a distributed file system that basically 'maps' data wherever it is located on a cluster. The tools for data processing are often on the same servers where the data is located, resulting in much faster data processing. If you're dealing with large volumes of unstructured data, Hadoop is able to efficiently process terabytes of data in just minutes, and petabytes in hours.

5) Resilient to failure:

A key advantage of using Hadoop is its fault tolerance. When data is sent to an individual node, that data is also replicated to other nodes in the cluster, which means that in the event of failure, there is another copy available for use.

Use:

Works on top of MongoDB. Retrieves tweets and users information from MongoDB and performs distributed parallel execution operations on different master/slave nodes of MongoDB.

Setup:

1) Prerequisite:

Execute the following commands:

```
cd /usr/lib/jvm
```

```
ln -s java-7-openjdk-amd64 jdk
```

```
sudo apt-get install openssh-server
```

2) Add Hadoop group and User:

```
sudo addgroup hadoop
```

```
sudo adduser --ingroup hadoop hduser
```

```
sudo adduser hduser sudo
```

After user is created, re-login into ubuntu using hduser

3) Setup SSH certificate:

```
ssh-keygen -t rsa -P ""
```

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

```
ssh localhost
```

4) Download Hadoop:

Download the compressed file for hadoop-0.20.0 from:

<http://apache.osuosl.org/hadoop/common/>

Extract the file 'hadoop-0.20.0.tar.gz' and place the 'hadoop' folder in '/usr/local/'.

Change ownership of folder.

```
sudo chown -R hduser:hadoop hadoop
```

5) Edit the .bashrc file for environment variable paths and add the following:

```
#Hadoop variables
```

```
export JAVA_HOME=/usr/lib/jvm/jdk/
```

```
export HADOOP_INSTALL=/usr/local/hadoop
```

```
export PATH=$PATH:$HADOOP_INSTALL/bin
```

```
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
```

```
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
```

```
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
```

```
export YARN_HOME=$HADOOP_INSTALL
```

Relogin and check that hadoop can then be accessed using terminal.

6) Change the contents of the following config files inside '/usr/local/hadoop/conf':

a) core-site.xml:

```
<?xml version="1.0"?>
```

```
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
```

```
<!-- Put site-specific property overrides in this file. -->
```

```
<configuration>
```

```
<property>
```

```
  <name>fs.default.name</name>
```

```
  <value>hdfs://localhost:9000</value>
```

```
</property>  
</configuration>
```

b) hdfs-site.xml:

```
<?xml version="1.0"?>  
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  
  
<!-- Put site-specific property overrides in this file. -->  
  
<configuration>  
  
<property>  
  <name>dfs.replication</name>  
  <value>1</value>  
</property>  
<property>  
  <name>dfs.namenode.name.dir</name>  
  <value>file:/home/hduser/mydata/hdfs/namenode</value>  
</property>  
<property>  
  <name>dfs.datanode.data.dir</name>  
  <value>file:/home/hduser/mydata/hdfs/datanode</value>  
</property>  
<property>  
  <name>hadoop.tmp.dir</name>  
  <value>/home/hduser/mydata/tmp</value>  
</property>  
  
</configuration>
```


c) mapred-site.xml:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>mapred.job.tracker</name>
    <value>localhost:9001</value>
  </property>
  <property>
    <name>mapred.task.tracker</name>
    <value>localhost:50060</value>
  </property>
</configuration>
```

d) Add to hadoop-env.sh:

```
export JAVA_HOME=/usr/lib/jvm/jdk/
```

7) Then place the following jar files inside hadoop's lib folder for integration of hadoop with mongodb:

mongo-2.7.3.jar, mongo-hadoop_0.20.205.0-1.1.0.jar, mongo-hadoop-core_0.20.205.0-1.1.0.jar, mongo-hadoop-pig_0.20.205.0-1.1.0.jar.

8) Then format the namenode by running command:

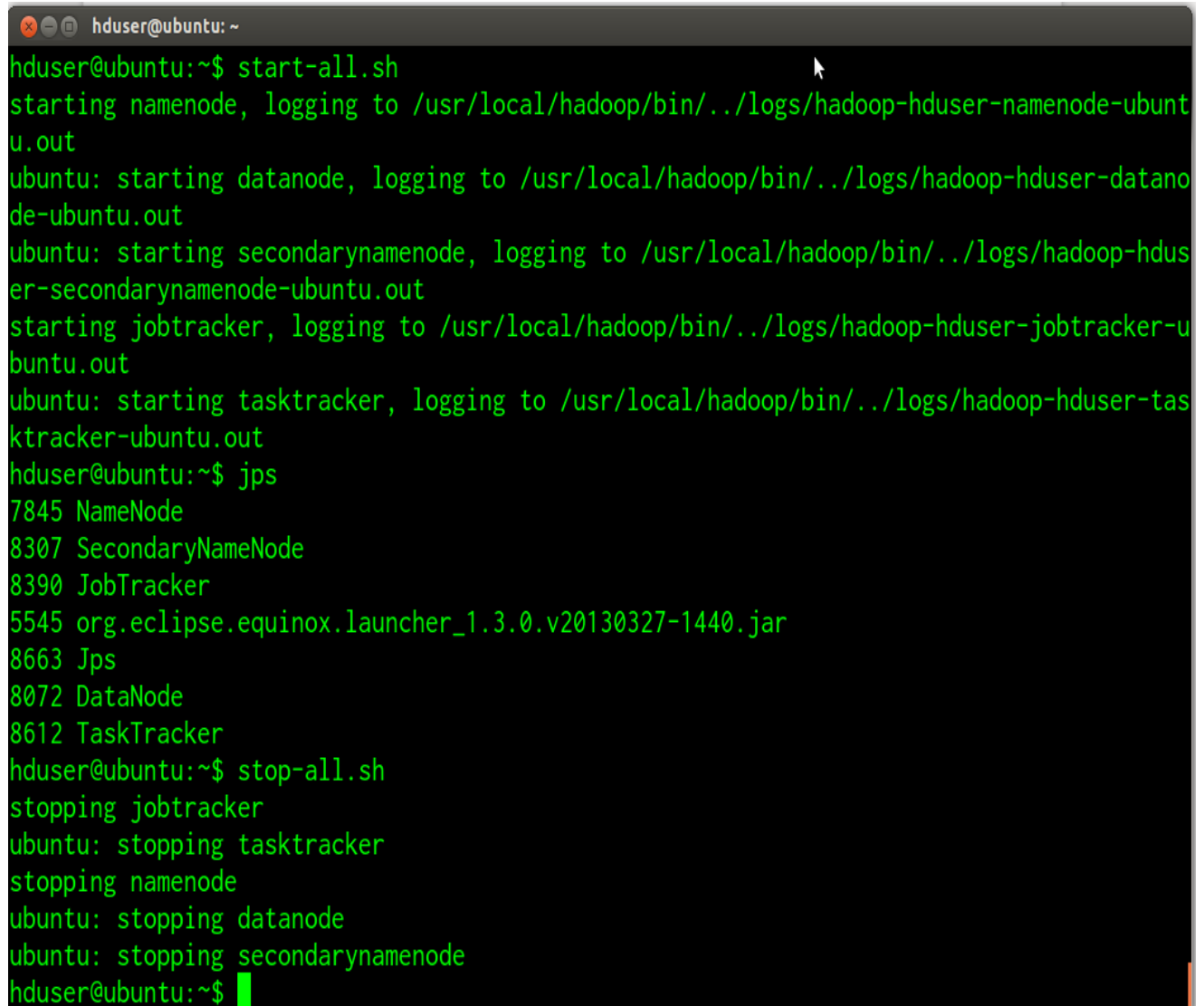
```
hdfs namenode -format
```

9) And then start hadoop by running command:

```
start-all.sh
```

10) Check if all processes are running using command:

jps



```
hduser@ubuntu: ~  
hduser@ubuntu:~$ start-all.sh  
starting namenode, logging to /usr/local/hadoop/bin/../logs/hadoop-hduser-namenode-ubuntu.out  
ubuntu: starting datanode, logging to /usr/local/hadoop/bin/../logs/hadoop-hduser-datanode-ubuntu.out  
ubuntu: starting secondarynamenode, logging to /usr/local/hadoop/bin/../logs/hadoop-hduser-secondarynamenode-ubuntu.out  
starting jobtracker, logging to /usr/local/hadoop/bin/../logs/hadoop-hduser-jobtracker-ubuntu.out  
ubuntu: starting tasktracker, logging to /usr/local/hadoop/bin/../logs/hadoop-hduser-tasktracker-ubuntu.out  
hduser@ubuntu:~$ jps  
7845 NameNode  
8307 SecondaryNameNode  
8390 JobTracker  
5545 org.eclipse.equinox.launcher_1.3.0.v20130327-1440.jar  
8663 Jps  
8072 DataNode  
8612 TaskTracker  
hduser@ubuntu:~$ stop-all.sh  
stopping jobtracker  
ubuntu: stopping tasktracker  
stopping namenode  
ubuntu: stopping datanode  
ubuntu: stopping secondarynamenode  
hduser@ubuntu:~$
```

6.4 Programming Tool (High-Level Platform): Pig Latin

Pig is a high-level platform for creating MapReduce programs used with Hadoop. The language for this platform is called Pig Latin. Pig Latin abstracts the programming from the Java MapReduce idiom into a notation which makes MapReduce programming high level, similar to that of SQL for RDBMS systems. Pig Latin can be extended using UDF (User

Defined Functions) which the user can write in Java, Python, JavaScript, Ruby or Groovy and then call directly from the language.

Pig vs SQL:

In comparison to SQL, Pig

- uses lazy evaluation,
- uses ETL,
- is able to store data at any point during a pipeline,
- declares execution plans,
- supports pipeline splits.

Pig Latin is procedural and fits very naturally in the pipeline paradigm while SQL is instead declarative. In SQL users can specify that data from two tables must be joined, but not what join implementation to use (You can specify the implementation of JOIN in SQL, thus "... for many SQL applications the query writer may not have enough knowledge of the data or enough expertise to specify an appropriate join algorithm.").

SQL is oriented around queries that produce a single result. SQL handles trees naturally, but has no built in mechanism for splitting a data processing stream and applying different operators to each sub-stream. Pig Latin script describes a directed acyclic graph (DAG) rather than a pipeline.

Reasons for using Pig:

1) It's quick

Pig's multi-query approach combines certain types of operations together in a single pipeline, reducing the number of times data is scanned. This means 1/20th the lines of code and 1/16th the development time when compared to writing raw MapReduce.

2) It supports any type of data

Pig got its name because it's omnivorous – it will happily consume any data you feed it: structured, semi-structured, or unstructured.

3) This pig does more with less.

Pig provides the common data operations (filters, joins, ordering, etc.) and nested data types (e.g. tuples, bags, and maps) missing from MapReduce.

4) Pig is easy to use.

It's easy to learn (especially if you're familiar with SQL) and opens Hadoop to data professionals who may not be software engineers.

5) It allows User Defined Functions

Pig is easily extensible by UDFs – including Python, Java, JavaScript, and Ruby – so you can use them to load, aggregate, or do sophisticated analysis.

6) It avoids breaking your jobs

Pig insulates your code from changes to the Hadoop Java API, so your jobs won't suddenly break due to an update. It also manages all details of submitting jobs and running complex data flows.

Use:

Works on top of Hadoop. Pig scripts retrieve tweets and users information from MongoDB and perform FILTER, JOIN and GROUP on them as required. The Pig script is automatically converted to MapReduce code which is executed by Hadoop.

The Pig script is embedded inside Java and the results are manipulated in Java and the first level summary is performed.

Setup:

Download the compressed file for pig from:

<http://www.apache.org/dist/pig/pig-0.12.0/pig-0.12.0.tar.gz>

Extract the file and place the pig folder 'pig-0.12.0' in '/usr/local/'.

Change ownership of folder.

```
sudo chown -R hduser:pig pig
```

Edit the .bashrc file for environment variable paths and add the following:

```
export PIG_HOME=/usr/local/pig-0.12.0
```

```
export PATH=$PATH:$PIG_HOME/bin
```

Relogin and pig can then be accessed using terminal.

Implementation:

Create a java project in eclipse and import the required jars:

java-json.jar, ejml-0.23.jar, hadoop-0.20.0-core.jar, commons-logging-1.0.4.jar, commons-logging-api-1.0.4.jar, log4j-1.2.17.jar, commons-httpclient-3.0.1.jar, pig-0.12.0-withouthadoop.jar, commons-codec-1.8.jar, commons-cli-1.2.jar.

Create an xml folder inside the project and then add the file in '/usr/local/hadoop/conf/core-site.xml' into it.

Also add 'shows.xml' and 'characters.xml' for getting the names of shows and characters.

The project is ready to contain java programs with embedded pig.

PigServer object is used to run the pig scripts inside java.

```
Properties props = new Properties();
```

```
props.setProperty("fs.default.name", "hdfs://localhost:9000");
```

```
props.setProperty("mapred.job.tracker", "localhost:9001");
```

```
PigServer pigServer = new PigServer(ExecType.MAPREDUCE,props);
```

Add the following external jars required by the pig script:

mongo-2.7.3.jar, commons-lang-2.6.jar, mongo-hadoop-core_0.20.205.0-1.1.0.jar, mongo-hadoop-pig_0.20.205.0-1.1.0.jar, mysql-connector-java-5.1.29.jar, joda-time.jar, pig-0.12.0.jar, piggybank-0.12.0.jar

Using function: `pigServer.registerJar("/path/file.jar");`

`pigServer.registerQuery("pig_query;");` is used to execute a query of the pigscript.

`pigServer.shutdown();` is used to close the pigServer.

Pig script is run for each character and each show in the xml files so that complete analysis is performed.

6.5 Database Connector: JDBC

JDBC is a Java-based data access technology (Java Standard Edition platform) from Oracle Corporation. This technology is an API for the Java programming language that defines how a client may access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases. A JDBC-to-ODBC bridge enables connections to any ODBC-accessible data source in the JVM host environment.

Statement – the statement is sent to the database server each and every time.

PreparedStatement – the statement is cached and then the execution path is pre-determined on the database server allowing it to be executed multiple times in an efficient manner.

CallableStatement – used for executing stored procedures on the database.

Update statements such as INSERT, UPDATE and DELETE return an update count that indicates how many rows were affected in the database. These statements do not return any other information.

Reasons for using JDBC:

- 1) Can read any database if proper drivers are installed.
- 2) Creates XML structure of data from database automatically
- 3) No content conversion required
- 4) Query and Stored procedure supported.
- 5) Can be used for both Synchronous and Asynchronous processing.
- 6) Supports modules

Use:

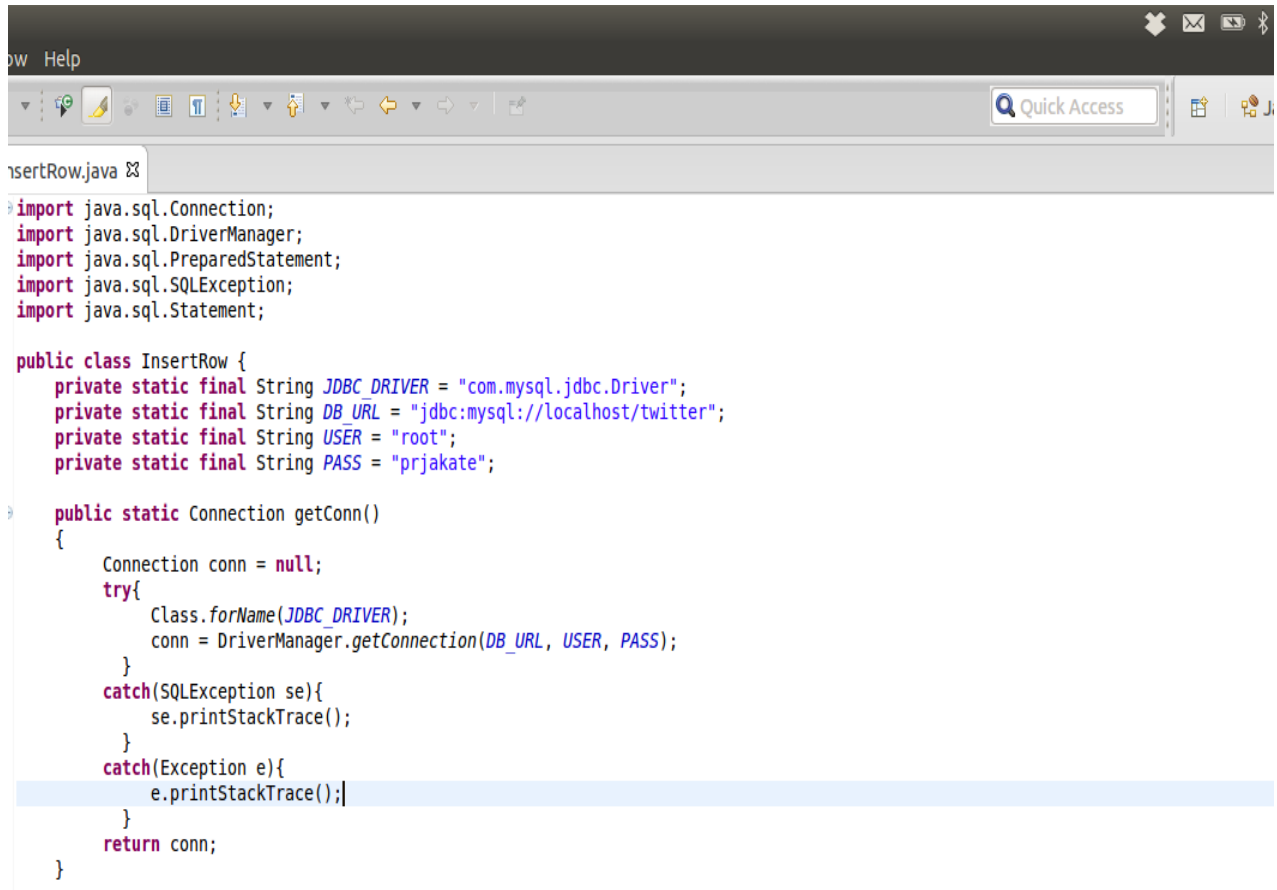
The results of the first level summary are stored inside MySQL database using JDBC drivers.

Implementation:

import the jar file:
mysql-connector-java-5.1.29.jar into the project.

Then create a connection to MySQL database by using the driver, URL, username and password as shown.

Use PreparedStatement/CreateStatement to perform queries on the database.



```
InsertRow.java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.sql.Statement;

public class InsertRow {
    private static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    private static final String DB_URL = "jdbc:mysql://localhost/twitter";
    private static final String USER = "root";
    private static final String PASS = "prjakate";

    public static Connection getConn()
    {
        Connection conn = null;
        try{
            Class.forName(JDBC_DRIVER);
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
        }
        catch(SQLException se){
            se.printStackTrace();
        }
        catch(Exception e){
            e.printStackTrace();
        }
        return conn;
    }
}
```

6.6 Working of The Analyser:

1) Input from Console:

Start time of summary, Path for Pig references.

2) Input from XML files:

Characters.xml -> Name of all characters

Shows.xml -> Names of all shows

3) Prepare Pig Server:

Create properties object and set the values of the properties "fs.default.name" to "hdfs://localhost:9000" and "mapred.job.tracker" to "localhost:9001".

```
Properties props = new Properties();
```

```
props.setProperty("fs.default.name", "hdfs://localhost:9000");
```

```
props.setProperty("mapred.job.tracker", "localhost:9001");
```

Create PigServer object and set the execution type to MAPREDUCE and set the properties using previously created object.

```
PigServer pigServer = new PigServer(ExecType.MAPREDUCE,props);
```

4) Register Pig Jars:

Use registerJar("path/file.jar") function of PigServer to register jars.

```
pigServer.registerJar(pig_path+"mongo-2.7.3.jar");
```

5) Pig Script:

Pig Script is executed by registering the query using registerQuery("query") function of PigServer. Firstly the pig script retrieves both the tweets and users collections from MongoDB.

```
pigServer.registerQuery("ret_tweets = LOAD 'mongodb://127.0.0.1:27017/twitter.tweets'
USING com.mongodb.hadoop.pig.MongoLoader('user_id, show_list, character_list, device,
inserted_at, sentiment, retweet_count, favorite_count, created_at') AS (user_id:chararray,
show_list:chararray, character_list:chararray, device:chararray, inserted_at:chararray,
sentiment:chararray, retweet_count:long, favorite_count:long, created_at:chararray);");
```

```
pigServer.registerQuery("users = LOAD 'mongodb://127.0.0.1:27017/twitter.users' USING  
com.mongodb.hadoop.pig.MongoLoader('id_str, gender') AS (id_str:chararray,  
gender:chararray);");
```

Then it checks if any tweets are present. If tweets are present then it performs JOIN on tweets and users collection.

```
pigServer.registerQuery("tweet_user = JOIN ret_tweets BY user_id, users BY id_str;");
```

Then for every show, it performs FILTER on show_list by using show name and for every character it performs FILTER on character_list using character name.

```
pigServer.registerQuery("tweets = FILTER tweet_user BY '"+type+"_list MATCHES  
'.*"+name+".*';");
```

The tweets are grouped according to those that were created in the same hour. The tweets are then counted on the basis of several conditions that is pos, neg, neu, male, female, male:pos, male:neg, male:neu, female:pos, female:neg, female:neu. The favorites and retweets are counted as well.

These counts are stored in the form of a Map whose key is the value of created_at corresponding to the beginning of an hour. After analysis on the basis of a show/character name, the Map is sent to a function that uses JDBC to insert the summary information in the tvshow/tvcharacter table of MySQL database.

Once such procedure is completed for all shows and characters, then the latest_time table is updated with latest time of summary.

Chapter 7

User interface

7.1 HTML

HTML is a language for describing web pages.

- HTML stands for Hyper Text Markup Language
- HTML is a markup language
- A markup language is a set of markup tags
- The tags describe document content
- HTML documents contain HTML tags and plain text
- HTML documents are also called web pages

HTML Tags

HTML markup tags are usually called HTML tags

- HTML tags are keywords (tag names) surrounded by angle brackets like `<html>`
- HTML tags normally come in pairs like `` and ``
- The first tag in a pair is the start tag, the second tag is the end tag
- The end tag is written like the start tag, with a forward slash before the tag name
- Start and end tags are also called opening tags and closing tags

`<tagname>content</tagname>`

HTML Elements

"HTML tags" and "HTML elements" are often used to describe the same thing.

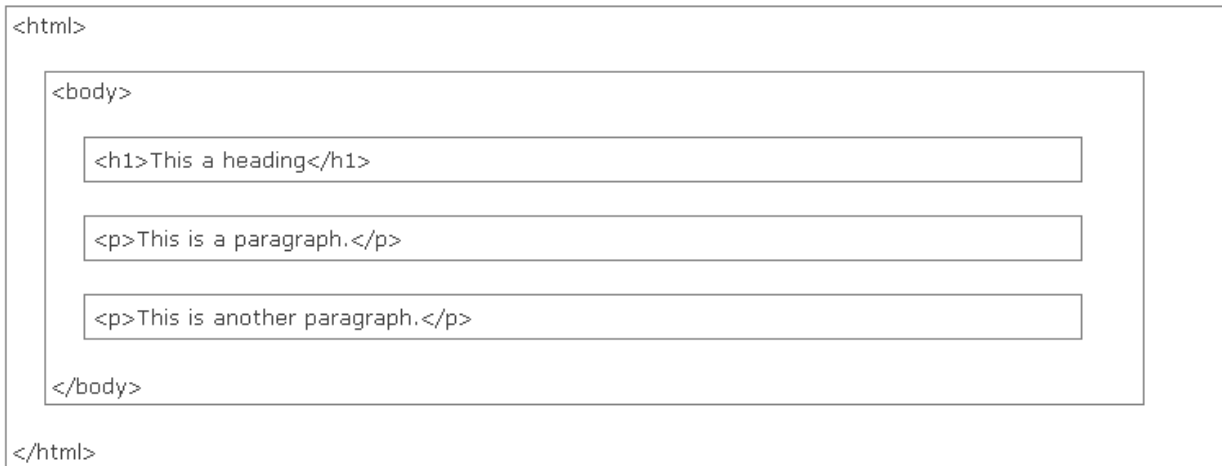
But strictly speaking, an HTML element is everything between the start tag and the end tag, including the tags:

HTML Element:

`<p>This is a paragraph.</p>`

HTML Page Structure

Below is a visualization of an HTML page structure:



Project Usage:

All the web pages that are designed are done with the help of HTML , HTML 5 Elements and Tags.

7.2 CSS

What is CSS?

- CSS stands for Cascading Style Sheets
- Styles define how to display HTML elements
- Styles were added to HTML 4.0 to solve a problem
- External Style Sheets can save a lot of work
- External Style Sheets are stored in CSS files

Styles Solved a Big Problem

HTML was never intended to contain tags for formatting a document.

HTML was intended to define the content of a document, like:

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

When tags like ``, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

In HTML 4.0, all formatting could be removed from the HTML document, and stored in a separate CSS file.

All browsers support CSS today.

CSS Saves a Lot of Work!

CSS defines HOW HTML elements are to be displayed.

Styles are normally saved in external .css files. External style sheets enable you to change the appearance and layout of all the pages in a Web site, just by editing one single file!

Reasons For Using CSS:

1. Separation of Content and Presentation.

Cascading Style Sheets are generally located in files separate from the main code (html, for example), permitting a team's different members, such as programmer and designer, to

focus on their specialties while working alongside each other, thereby avoiding the risk of interfering with each other's work and affecting the final product.

2. Flexibility.

With No effect on the content , the styling of the web page can be changed.

3. Consistency of page design.

A website, whether it is dynamic or static, is usually comprised of numerous pages. Maintaining a single, consistent appearance can become a difficult and tedious task if you must copy and paste code each time you create a new page, or want to modify a single aspect across the entire site. Cascading style sheets link all the pages of a website, speeding up this process and minimizing the work load.

4. Optimization of Load Time, Server Traffic and Content to Code Ratio.

After having separated a website's design and content, the file size is significantly reduced. It is not unusual to see reductions of 50% after switching from tables to CSS. The primary explanation for this dramatic decrease is that presentation information is placed in the external CSS document, called up once when the homepage loads up and then cached (stored) on to the user's computer. Alternatively, table layouts place all presentation information inside each HTML, which is then called up and downloaded for every page on the site. Smaller files provide three benefits. Firstly, you noticeably reduce the time it takes the site to load in the browser. Secondly, you reduce bandwidth costs, which for high traffic sites can mean enormous savings. Lastly, your site will rank higher in the SERP's due to a higher content to code ratio.

5. Precision or Elasticity.

By utilizing CSS, you can specify the exact size and positioning of the elements that form your pages, including in which pixel the browser should place this or that image, and how high and wide it should be. At the same time, you can use a host of different measures that permit you to expand or contract the content within the navigational window, independent of the user's screen resolution.

6. Increase Compatibility.

The more surfers you can allow will inevitably lead to an increase in web traffic and eventually conversions. A CSS-based website is compatible with PDA's, mobile phones, in-car browsers and Web TV. Also, unlike a tabular layout, you can make an additional CSS document specifically for handheld devices, which will be called up in place of the regular CSS document, thereby ensuring your website is accessible to this lucrative market.

7. Clean Source Code.

If you write an independent style sheet, the page's source code will be less confusing, in addition to speeding up the process of line placement. Also, a clean code is more accessible to search engines, thereby improving their ability to spider your content, leading to higher rankings in the SERP's.(Search Engine Result Page).

8. Compatibility and Continuity.

The rules established by the CSS-1 specification fixed the design standards, which are maintained and respected in CSS-2. We can assume that the same will be true of CSS-3 as was of its predecessor. Interestingly, browsers that do not support CSS-3 will not encounter issues when assimilating CSS content since it will always be compatible with CSS-2 or CSS-1. Compatibility with previous CSS specifications will always be guaranteed.

Project Usage:

The styling is provided to the Elements and Tags using external style sheets which are imported within a particular web page as well as using them along with the tags wherever it was required. Bootstrap CSS is used to a great extent to reduce the styling work and to make the website a responsive one adjusting itself according to the Screen dimensions.

7.3 JAVACRIPT

JavaScript is a Scripting Language

JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that display. JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

Reason for using Javascript

JavaScript is an excellent solution to implement when validating input forms on the client side. This means that if a user forgets to enter his name in a form for instance a JavaScript validation function can popup a message to let him know about the omission. This is a far better solution than having a server side validation routine handle the error because the server does not have to do any additional processing. An asp or php routine could be written to achieve the same task but the JavaScript would not allow the form to be submitted unless it was completed properly in the first place, a much more robust solution!

Another area where JavaScript excels is in the creation of dynamic effects such as rollover images and scripted slideshows, where its use has become commonplace. Because JavaScript runs inside the client's browser it can be used to change the appearance of the user's screen after the page has been sent by the server. This allows it to create some very impressive dynamic image effects.

Project Usage:

- For sending parameters to the next Web Page
- For Displaying the data on the page depending upon the user input provided by the user.
- To alert the user when he/she is trying to do something that prohibited.
- A lot of Google Charts Scripts are used and the data is provided to this charts using javascript and php.
- The Google charts used are as follows:
 - 2-D Pie Chart
 - Donut Chart
 - 3-D Pie Chart
 - Bar Chart

7.4 JQUERY

What is jQuery?

jQuery is a lightweight, "write less, do more", JavaScript library.

The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

- HTML/DOM manipulation

- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

Reasons for using JQuery:

1. jQuery promotes simplicity

Developers find jQuery intuitive and easy to learn -- this library is built on shorter, simpler code, after all. With simple syntax and open coding standards, developers can shorten the time it takes to deploy an application or site.

In addition, developers don't have to be experts in programming or Web design to create great styles for their sites. Any developer who has spent hours coding and testing CSS files will surely appreciate the simple implementation that jQuery brings to the table. There's also a set of robust jQuery UI components that developers can plug into their websites.

2. jQuery elements display even when JavaScript is disabled

If Adobe Flash isn't installed on any given browser, certain parts of the page may render incorrectly, if they render at all. This is not only unpleasant for the user; it forces developers to spend extra time coding for the browsers that lack the Flash plug-in, which adds to development time.

Not so with jQuery. Manipulating the HTML DOM has become the most widely accepted practice of manipulating a Web page so content will be rendered even if JavaScript is disabled in the browser. Since the HTML DOM is always present, there's no more worrying about browser settings either.

Furthermore, developing using jQuery can reduce instances of help desk tickets. Your help desk will appreciate that your developers are coding proactively to avoid dreaded browser crashes.

3. jQuery pages load faster

Google and other search engines using page load time as one of the many factors affecting SEO. (More on that later.) For this, and many other, reasons, every developer should strive to make code as light and concise as possible.

The best way to do this is to reduce the size of your code. If your site is coded with an HTML and CSS base, you can easily make uniform adjustments to your code that will reduce the size. Like CSS, jQuery files are generally stored separately from the Web page itself. This lets developers make modifications across the entire site through one central repository instead of search through folder structures. This is a core benefit of CSS coding, and it is a proven success.

In addition, jQuery gives you the option of loading div tags only when you need them. If you are taking measures to improve the speed of your website, then you may consider loading only the necessary div tags needed for your page load event. This way, you can display only what a user needs to see right away and have the rest of your division elements load as they are needed.

4. jQuery can be SEO friendly

You may have the most appealing site, but is it worth it if you sacrifice style for SEO? The way you code your site greatly affects the way it can be found in Google, Bing, and other search engines.

As noted, jQuery can be optimized for search engines, and there are a lot of plug-ins available to aid developers in this task. Embedding your jQuery elements using unordered lists is an SEO-friendly practice that works well.

Another SEO advantage of the HTML5-jQuery combo that's worth mentioning is that animations can be loaded with keywords that can be read by search engines.

Project Usage:

- Since we have incorporated the use of Bootstrap into our project, it provides a lot of JQuery features that are used in the project.
- Also for inculcating the use of Google Charts , it was mandatory to include the jQuery of Google API's.

7.5 PHP

What is PHP?

- PHP is an acronym for "PHP Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP costs nothing, it is free to download and use

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

What Can PHP Do?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can restrict users to access some pages on your website
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free.

Reasons for using PHP:

1. Ease of use
2. Integrates with HTML, CSS, javascript, ajax, jquery very well.
3. Well documented
6. Database communication is excellent

8. Compatibility across multiple platforms and browsers

Project Usage:

- For receiving the parameters sent by the previous webpage using “GET” method.
- For maintaining the Session Variables.
- To output text on the web page using the if-else conditions and the php variables.
- Connecting the MySql database.
- Populating Google Charts with the data extracted from database using php.
- For **including** the external php files within a web page.
- To alter the way a web page is loaded thereby providing the dynamic web pages.
- Making use of php variables within javascript.

7.6 BOOTSTRAP

Bootstrap is a free collection of tools for creating websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. It is the No.1 project on GitHub .

Features:

Bootstrap is compatible with the latest versions of all major browsers. It gracefully degrades when used on older browsers such as Internet Explorer 8.

Since version 2.0 it also supports responsive design. This means the layout of web pages adjusts dynamically, taking into account the characteristics of the device used (desktop, tablet, mobile phone).

Bootstrap is open source and available on GitHub. Developers are encouraged to participate in the project and make their own contributions to the platform.

Reasons for using Bootstrap:

- **Easy to get started**

CSS Pre-processing is great and every front end developer should learn it. However not everyone is using it. There are still many designers creating and managing CSS files the same old way. Bootstrap offers LESS files for those of us who know how to use it, but it also provides the plain old CSS file for those don't want to use CSS pre-processing.

To take advantage of what Bootstrap has to offer, you just have to download the files from Bootstrap on Github and after unzipping, include the files in the head of your HTML document.

This example HTML document includes the bootstrap framework with its default styling and every single components and JavaScript plugins.

- **Great grid system**

Bootstrap is built on responsive 12-column grids, layouts and components. Whether you need a fixed grid or a responsive, its only matter of a few changes. Offsetting & Nesting of columns is also possible in both fixed and fluid width layouts.

Another useful set of features are the responsive utility classes using which you can make a certain block of content appear or hide only on devices based on the size of their screen. Very handy when you want to hide some content based on screen size. Adding a class such

as `.visible-desktop` to a element, will make it visible only for desktop users. There are similar classes for tablets and phones.

- **Base styling for most HTML elements**

A website has many different elements such as headings, lists, tables, buttons, forms, etc. All these fundamental HTML elements have been styled and enhanced with extensible classes.

The HTML elements for which styles are provided are:

- Typography
- Code
- Tables
- Forms
- Buttons
- Images
- Icons

- **Extensive list of components**

Whether you need drop down menus, pagination or alert boxes, Bootstrap has got your covered. Styling of every single element follows a consistent theme and if you know LESS, then customizing it takes just few minutes.

Some of the components pre styled are:

- Dropdowns
- Button Groups
- Navigation Bar
- Breadcrumbs
- Labels & Badges
- Alerts
- Progress Bar

- And many others.
 - **Bundled Javascript plugins**
-

The components such as drop down menu are made interactive with the numerous JavaScript plugins bundled in the bootstrap package.

If your project requires sliders, tabs, accordions, then you no longer have to try and test numerous different plugins across the web. Adding these functionalities is just a matter of adding few lines of code and you are all set. With the customization option you can also choose only certain plugins to keep the file size to a minimum.

- **Good documentation**
-

Not only does Bootstrap offer styling for almost every element a typical website or web application requires, it also provides a great documentation with examples and demo that only make it more easier for even someone new.

Project Usage:

Almost entire website is modeled on the Bootstrap Components especially the CSS components.

7.7 GOOGLE CHARTS

Google Charts provides a perfect way to visualize data on your website. From simple line charts to complex hierarchical tree maps, the chart gallery provides a large number of ready-to-use chart types.

The most common way to use Google Charts is with simple JavaScript that you embed in your web page. You load some Google Chart libraries, list the data to be charted, select options to customize your chart, and finally create a chart object with an id that you choose. Then, later in the web page, you create a <div> with that id to display the Google Chart.

Charts are exposed as JavaScript classes, and Google Charts provides many chart types for you to use. The default appearance will usually be all you need, and you can always customize a chart to fit the look and feel of your website. Charts are highly interactive and expose events that let you connect them to create complex dashboards or other experiences integrated with your webpage. Charts are rendered using HTML5/SVG technology to provide cross-browser compatibility (including VML for older IE versions) and cross platform portability to iPhones, iPads and Android. Your users will never have to mess with plugins or any software. If they have a web browser, they can see your charts.

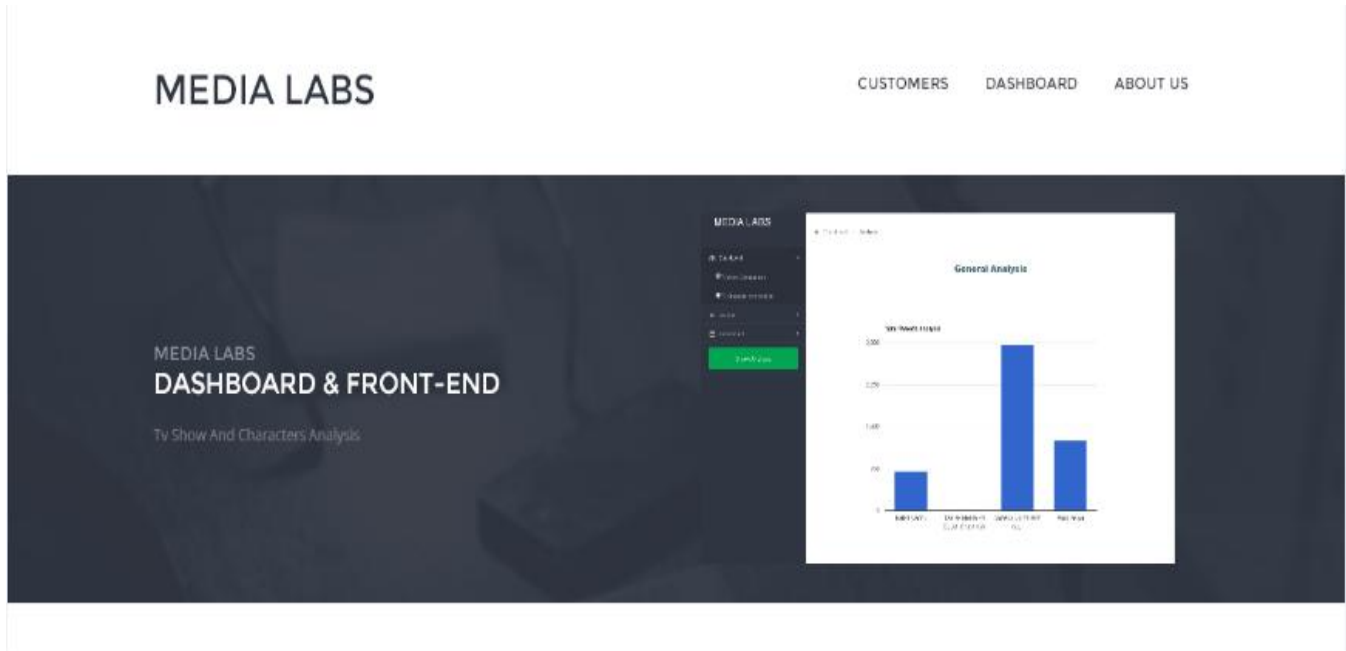
All chart types are populated with data using the DataTable class, making it easy to switch between chart types as you experiment to find the ideal appearance. The DataTable provides methods for sorting, modifying, and filtering data, and can be populated directly from your web page, a database.

Project Usage:

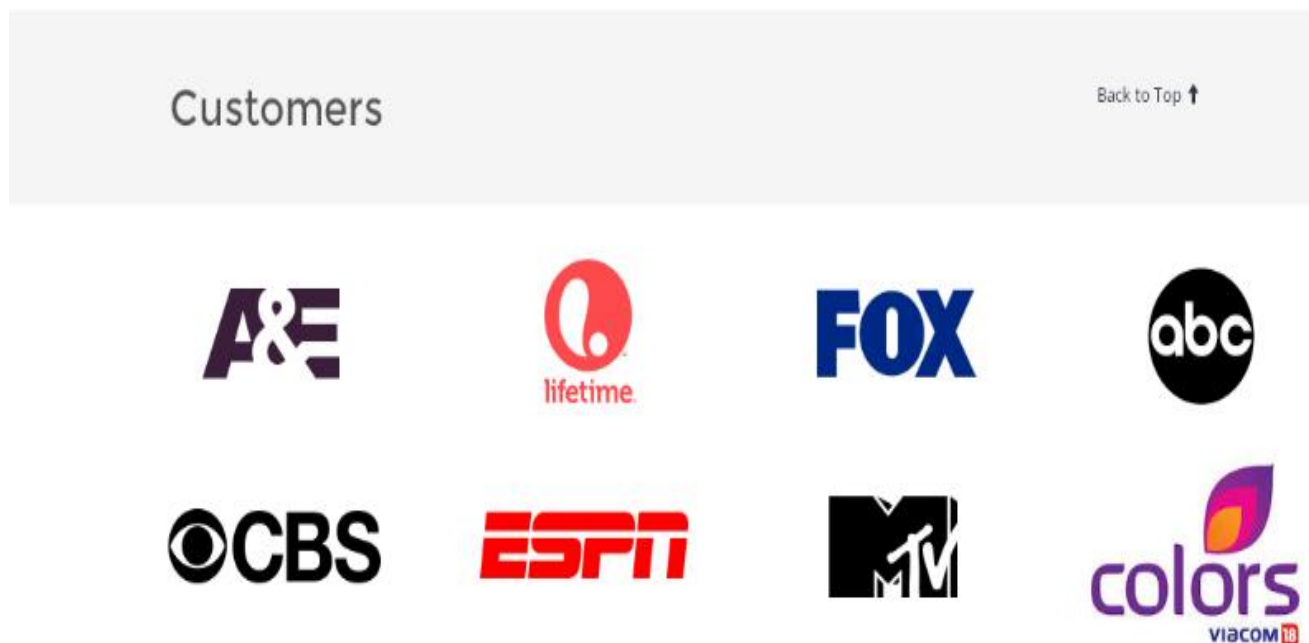
- For Sentiment Analysis , **Donut Charts** are used
- For Gender Analysis , **3-D Pie Charts** are used.
- For Device-Wise Analysis, **2-D Pie Charts** are used
- For General Analysis , **Bar Graph** and tabular representation is used.

7.8 Implementation

Home page



Customer page







Dashboard


MEDIA LABS

- Dashboard
- Analysis
 - Sentiment Analysis
 - Gender Analysis
 - Device-wise Analysis
- Time Period
- Show Analysis

Dashboard

Balika Vadhu

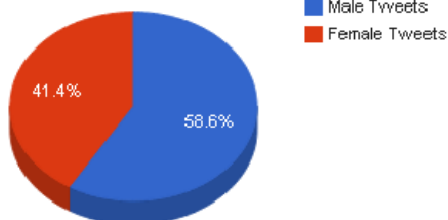







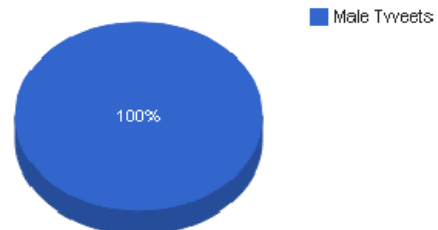
Gender wise analysis

Gender Analysis

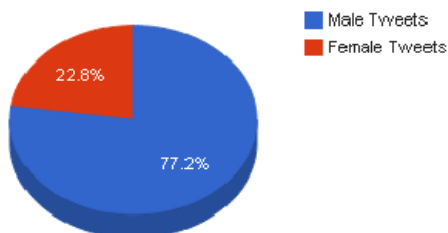
Balika Vadhu



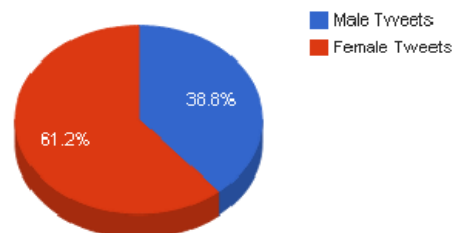
Tarak Mehta Ka Ooltah Chashmah



Comedy Nights With Kapil



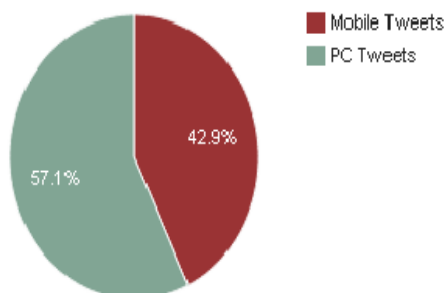
Rangrasiya



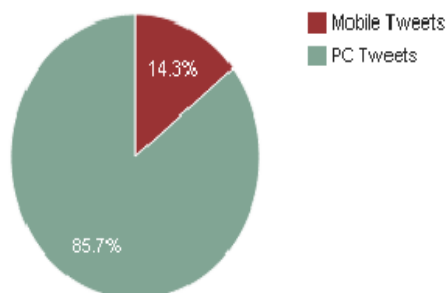
Device wise analysis

Device-wise Analysis

Balika Vadhu



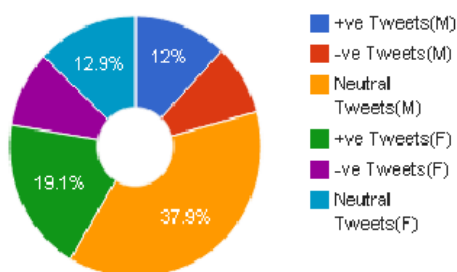
Taarak Mehta Ka Ooltah Chashmah



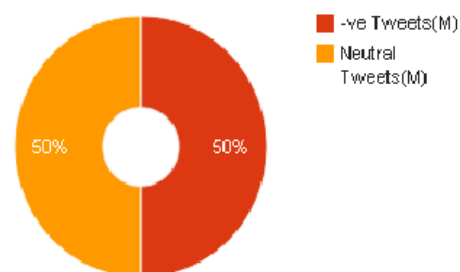
Sentiment analysis

Sentiment Analysis

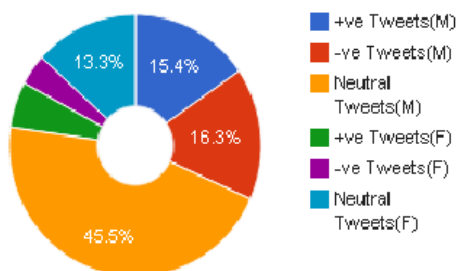
Balika Vadhu



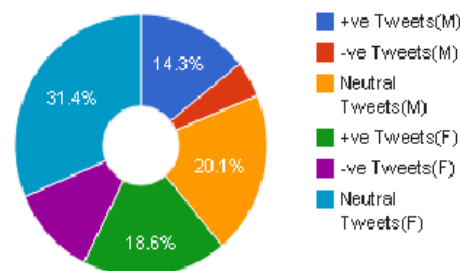
Taarak Mehta Ka Ooltah Chashmah



Comedy Nights With Kapil



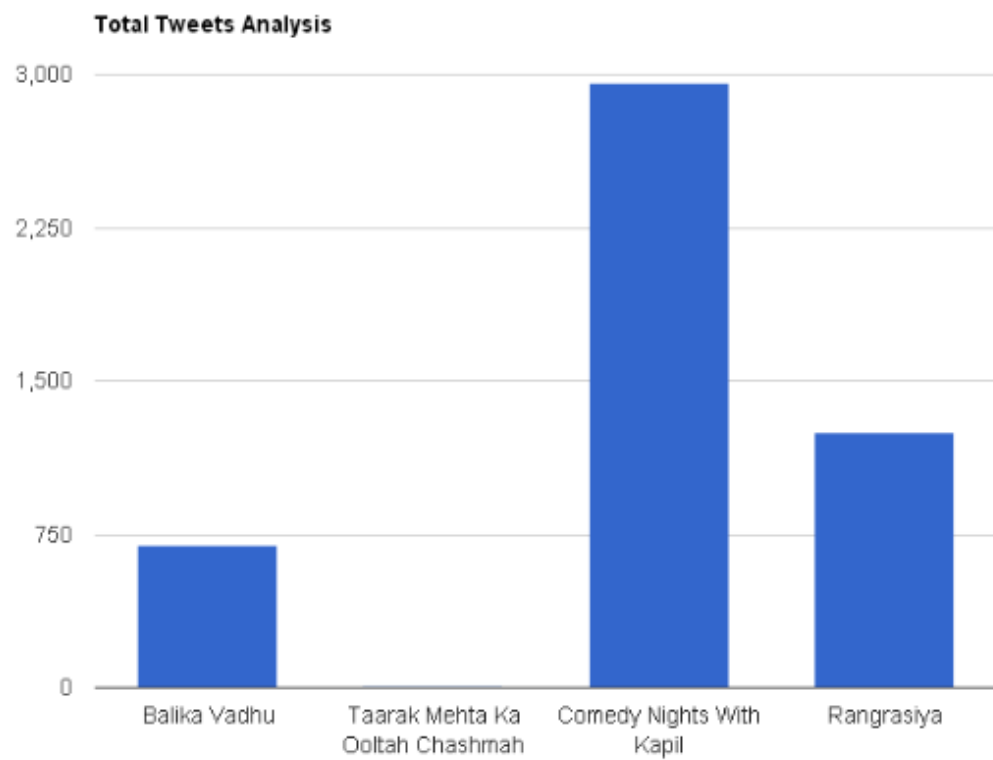
Rangrasiya



General analysis

🏠 Dashboard / Analysis

General Analysis



Chapter 8

Future work and Conclusion

8.1 Future work

1 Extend the Channels

In the current scenario we are just considering few shows which are either daily soaps, emotional drama or comic one but in future we can extend this application to get reviews of other Channels like Sports, news, tournaments (eg. IPL). Due to which we could use it in various field to get the more reviews from the people.

2 Extend for source

The only medium used in this case to fetch data is the twitter api. With the help of it the tweets are been analyzed and represented. In this case we can extend the source by fetching information from other social sites like facebook, quora where people express their concern regarding particular topic

3 More tweet

If we put the given application live on the AWS, we have a better chance of getting real time tweet which would be helpful while comparing some very recent event like any match.

4 Junk tweet

Many times it is observed that there are many tweet which are junk tweet.

So while performing the anlaysis of it we cannot get the exact outcome from the tweet, if we perform some process to remove it than we could get a very precise analysis of the given tweet.

5 Better gender api to get accurate gender prediction

8.2 Conclusion

In this application we considering twitter as a data source we fetch the tweet from in with the help of twitter api. Then the complete data is been analyze with the help of mongodb where it is processed. The processed data is then stored in the MYSQL database from where with the help of the PHP and UI it is represented to the client. This application thus help the advertising firm to get a complete review of the people and there will be no need for them to be dependent on the TRP constraint.

Chapter 9

References

Books

- 1) Hadoop The Definitive Guide O'Reilly and Yahoo press, author: Tom White
- 2) Programming Pig O'Reilly publications, author: Alan Gates
- 3) MongoDB The Definitive Guide O'Reilly publication, author: Kristina Chodorow

Papers

- 1) IEEE Hassan A., Abbasi, A., Zeng, D. 2013 Twitter Sentiment Analysis: A Bootstrap Ensemble Framework
- 2) IEEE Lima, A.C.E.S., de Castro, L.N. 2012 Automatic sentiment analysis of Twitter messages
- 3) IEEE Ying Zhu 2012 Introducing Google Chart Tools and Google Maps API in Data Visualization Courses
- 4) IEEE Singh, S. , Singh, N. 2012 Big Data analytics
- 5) IEEE Alan F. Gates, Jianyong Dai, Thejas Nair 2013 Apache Pig's Optimizer

Document Links

- 1) Twitter API docs <https://dev.twitter.com/docs>
- 2) Pig docs <http://pig.apache.org/docs/r0.12.0/index.html>
- 3) Hadoop docs <https://hadoop.apache.org/docs/r0.18.3/>
- 4) Twitter4j docs <http://twitter4j.org/javadoc/>
- 5) ACM Elif Dede, Madhusudhan Govindaraju, Daniel Gunter, Richard Shane Canon, Lavanya Ramakrishnan 2013 Performance evaluation of a MongoDB and hadoop platform for scientific data analysis.
- 6) Google Charts docs <https://developers.google.com/chart/interactive/docs/gallery/>

