BOX OFFICE REVENUE PREDICTION USING DUAL SENTIMENT ANALYSIS

A Project Report submitted in partial fulfilment of the requirements for award of the degree

BACHELOR OF ENGINEERING

In

COMPUTER ENGINEERING

Of

SAVITRIBAI PHULE PUNE UNIVERSITY

By

SHEFALI SINHA	**Seat No : B120234416**
PRIYANKA SAPKAL	**Seat No : B120234403**

Under the guidance of

PROF. D. D. GATADE
MR. PARESH SANGHANI

Department of Computer Engineering
Sinhgad College of Engineering

Vadgaon (Bk.), Pune-411041

2015-2016

CERTIFICATE

This is certified that the Project Report entitled

BOX OFFICE REVENUE PREDICTION USING DUAL SENTIMENT ANALYSIS

Submitted by

Shefali Sinha
Priyanka Sapkal

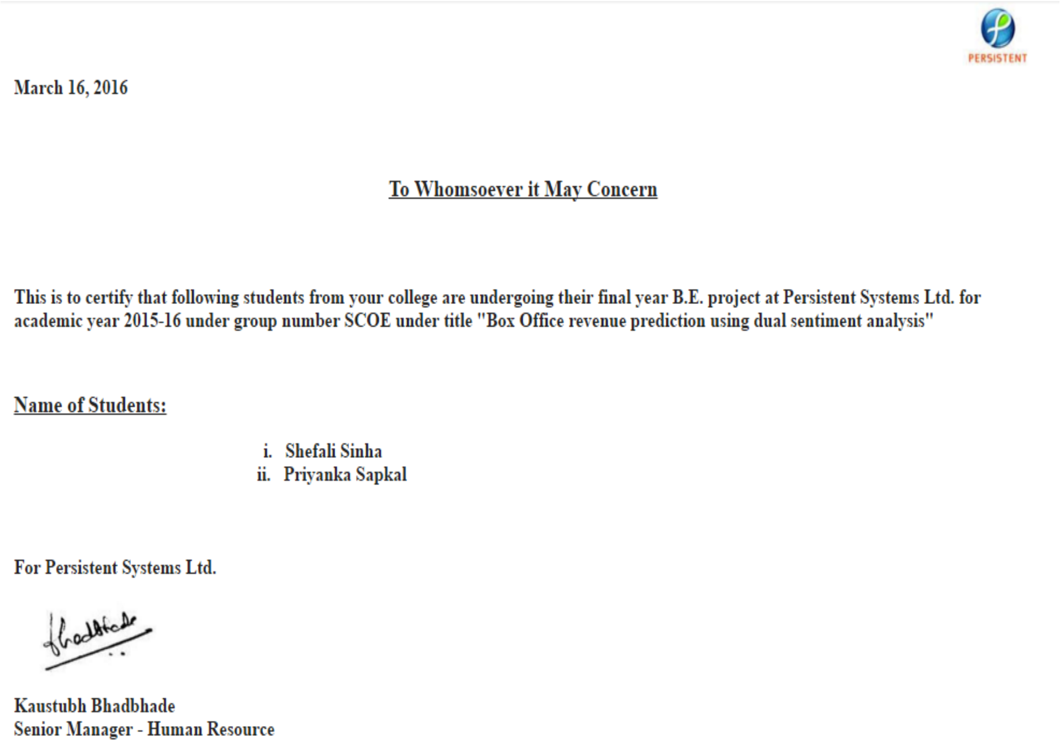
Have successfully completed their project under the supervision of Prof. D. D. Gatade and Mr. Paresh Sanghani for the partial fulfillment of Bachelor of Engineering in Computer Engineering of Savitribai Phule Pune University. This work has not been submitted elsewhere for any degree.

Prof. D. D. Gatade
Internal Guide

Prof. P. R. Futane
H.O.D.

Prof. S. D. Lokhande
Principal

Sponsorship Letter



PERSISTENT

March 16, 2016

To Whomsoever it May Concern

This is to certify that following students from your college are undergoing their final year B.E. project at Persistent Systems Ltd. for academic year 2015-16 under group number SCOE under title "Box Office revenue prediction using dual sentiment analysis"

Name of Students:

i. Shefali Sinha
ii. Priyanka Sapkal

For Persistent Systems Ltd.

Kaustubh Bhadbhade
Senior Manager - Human Resource

ACKNOWLEDGMENTS

We are grateful to **Prof. D. D. Gatade** and **Mr. Paresh Sanghani** for their expert guidance and continuous encouragement throughout this intricate part of the project.

We would like to express appreciation towards **Dr. S. D. Lokhande**, Principal, Sinhgad College of Engineering, Pune, **Prof. P. R. Futane**, Head of Department of Computer Engineering and our Project Coordinator whose invaluable guidance supported us in completing the project report of Box Office Revenue Prediction Using Dual Sentiment Analysis.

At last we must express our sincere gratitude to all the staff members of Computer Engineering Department who helped us directly or indirectly during this course of work.

Shefali Sinha
Priyanka Sapkal
(BE Comp. Engg.)

ABSTRACT

Today a large population uses Twitter and other such social networking websites. A lot of mass opinion is available on such sites. Success of a movie does not depend only on its content. Pre-release buzz, star-cast, budget etc are also important. Poster and music release, big promotions done for publicity instill excitement among the people which leads to increased number of tweets about the movie. We can use these tweets to examine what impact the movie had on the mob, and thus we can predict what will be the approximate revenue of the movie on release. In our project, we have used dual sentiment analysis to defer the polarity of a tweet as positive, negative or neutral. Then we consider several other factors such as, star-cast, holiday effect, other competitive movies, status of the last few movies of the actors, sequel, etc. We use regression model to predict the output based on all these parameters.

LIST OF FIGURES

LIST OF TABLES

ABBREVIATION

DSA	Dual Sentiment Analysis
BOW	Bag Of Words
UI	User Interface
PK	Primary Key
FK	Foreign Key
API	Application Programming Interface
UML	Unified Modelling Language

TABLE OF CONTENTS

Chapter-1
Introduction

1.1 Background and basics

Social networking websites are used by a large sector of people, of almost all age groups across the globe. People,ranging from Celebrities to a common man, use it to connect with other people and express their views on topics such as politics, economy, entertainment, etc. Social media is seen as a means of gathering insights into human behaviors. Twitter provides a wide platform for collecting mass opinion. Chatter from these sites can be used to make various predictions. For example, predictions of election result used such data.Similarly, movie success can also be predicted using the knowledge from these sites. As it is said, more the movie is talked about, more money it will make. Buzz on Twitter can be used to predict how a movie performs post its release.

Sentiment Analysis can be done on data extracted from social media ,the results of which , can be used to determine whether the people are in favor of or against a particular thing. Such knowledge proves usefull in predicting the success of a movie. Audience can also decide a movie that they will watch based on the prediction given using twitter data. But processing twitter data is difficult because of its ungrammatical structure. Bag-of-words is typically used for text classification. But its performance is limited. Hence,A simple yet efficient model is proposed, called dual sentiment analysis (DSA), to address the polarity shift problem in sentiment classication. There, we measure not only how positive/negative the original text is, but also how negative/positive the reversed text is.

1.2 Literature Survey

1.2.1 Sentiment Analysis

Significant amount of work is done in the field of data mining to extract the polarity of the text. However, they follow a very straightforward technique of sentiment analysis which sometimes leads to wrong interpretation of sentiments. One of the traditional method for sentiment classification is Bag-Of-Words model .

Bag-Of-Words Model

The Bag-Of-Words (BOW) model is typically used for text representation.

Over View

In the BOW model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and even word order but keeping multiplicity. The statistical machine learning algorithms (such as Naive Bayes, maximum entropy classifier, and support vector machines) are then employed to train a sentiment classifier. The bag-of-words model is commonly used in methods of document classification, where the (frequency of) occurrence of each word is used as a feature for training a classifier.

Methods to address Polarity Shift Problem

According to the levels of granularity, tasks in sentiment analysis can be divied into four categorizations: document-level, sentence-level, phrase-level, and aspect-level sentiment analysis. Focusing on sentence level, we have:
The term-counting methods can be easily modified to include polarity shift. One common way is to directly reverse the sentiment of polarity-shifted words, and then sum up the sentiment score word by word . Compared with term counting methods, the machine learning methods are more widely discussed in the sentiment classification literatures. However, it is difficult to integrate the polarity shift information into the BOW model in such methods. For example,using a method by simply attaching *NOT* to words in the scope of negation, so that in the text *I dont like book*, the word *like* becomes a new word *like NOT*. But method only has slightly negligible effects on improving the sentiment classification accuracy. For example, there are proposed models that do negation by looking for specific part-of-speech tag patterns or use syntactic parsing to capture three types of valence shifters (negative, intensifiers, and diminishers). Their results showed that handling polarity shift improves the performance of term-counting systems significantly, but the improvements upon the baselines of machine learning systems are very slight.

1.2.2 Shortcomings

Although the BOW model is very simple and quite efficient in topic-based text classification, it is actually not very suitable for sentiment classification because it disrupts the word order, breaks the syntactic structures, and discards some semantic information.
Due to the fundamental deficiencies in BOW, most of the efforts showed very slight effects in improving the classification accuracy. One of the most well-known difficulties is the polarity shift problem. Polarity shift is a kind of linguistic phenomenon which can reverse the sentiment polarity of the text. Negation is the most important type of polarity shift. For example, by adding a negation word dont to a positive text I like this book in front of the word like, the

sentiment of the text will be reversed from positive to negative. However, the two sentiment-opposite texts are considered to be very similar by the BOW representation. This is the main reason why standard machine learning algorithms often fail under the circumstance of polarity shift.

Several approaches have been proposed in the literature to address the polarity shift problem. However, most of them required either complex knowledge or extra human annotations. Such high-level dependency on external resources makes the systems difficult to be widely used in practice.

1.2.3 Prediction Applications

At present there are a lot of applications using various techniques for predicting the success of a movie. There are systems that mine through social media data to predict box office revenues of upcoming movies. There are systems that use the pre-release buzz and polarity of sentiments to figure out how successful the movie will be. But those systems follow a very straightforward technique of sentiment analysis which sometimes leads to wrong interpretation of users sentiments. There are also systems that take number of searches as the parameter, example, more the number of searches, more successful the movie will be. And also, these systems dont consider the additional information,such as, star cast, status of the last few movies of the actors, holiday effect,competitive movies, number of the times the tweet has been retweeted which reduce the performance of the system considerably.

1.3 Project Undertaken

In our project, we use Dual Sentiment Analysis (DSA) for overcoming the limitations of the traditional bag-of-words model and use it for efficient sentiment classification. The result of DSA on twitter data along with other static factors are further used to predict the Box Office Revenue of a movie.

1.3.1 Problem definition

The proposed project is a web based application that provides the Box Office Revenue prediction of movies to the audience and other concerned people through simple and user-friendly Graphical User Interface. Prediction is based on the polarity ratio of tweets calculated using DSA, hype, star cast of movie, its genre, sequel and holiday effect.

1.3.2 Scope Statement

Project Scope Description

Presently there are systems that mine through social media data to predict box office revenues of upcoming movies. There are systems that use the pre-release buzz and polarity of sentiments to figure out how successful the movie will be. But those systems follow a very straightforward technique of sentiment analysis which sometimes leads to wrong interpretation of users sentiments. There are also systems that take number of searches as the parameter, example, more the number of searches, more successful the movie will be. And also, these systems dont consider the additional information,such as, star cast, status of the last few movies of the actors, holiday effect,competitive movies, number of the times the tweet has been retweeted or favorited,etc. These all factors when taken into account, can lead to higher quality of output.

Hence, a system is proposed, that will perform multivariate regression considering all the above stated parameters. Moreover, a technique called Dual Sentiment Analysis is used on the Twitter data. Each tweet will be reversed and considering both things, i.e, how positive the original tweet is and how negative the reversed tweet is or how negative the original tweet is and how positive the reversed tweet is, the final decision will be made if the tweet falls in positive, negative or neutral category. The system, a user friendly web application, will show a list of the upcoming movies on the home page. When a user will click on any movie of his choice, he/she will be shown what is the approximate revenue predicted by our system. Along with this, he/she will also be given the list of all the factors that were considered for the prediction. The web page can also play the official trailer of the selected movie.

Project Justification

Early prediction of a movie can be very useful for the makers, distributors and the marketing unit of the movie. They can have an idea about how well the movie is going to perform. The makers can get an estimate of their income from the movie. The marketing unit can know what is the public talking about the movie, what are their opinions. Accordingly, they can plan their promotion activities. With appropriate promotion of the movie the number of audience might increase. The release of the music album or the poster of the movie usually creates the hype amongst the mob.All such activities can be planned pre-release, which will lead to better income post release. Audience can also pick a movie that they will watch based on the prediction given.

Project Deliverables

The application provides:

1. Revenue prediction of movies.

2. All the factors considered in calculating the final revenue.

When the movie will be out from the theatres, its actual revenue will be stored and the error factor, if any, will be considered in next prediction so as to increase the accuracy of the system. The application provides a simple Graphical User Interface to the user.

User Acceptance Criteria

The application should provide the approximate revenue of movies through simple and easy to use GUI . It should also provide the user, all the factors and their contribution in prediction.

Project Boundaries

The application uses twitter data as means of collecting mass opinion. Data from any other social media is not considered in this project.

Project Constraints

Following are some of the constraints to be kept in mind:

- Due to Twitter API limitations, only a small part of total tweets available can be caught.
- For more accurate predictions, more time needs to be given to the system.
- Tweets in English only can be used, as our system will be totally supported in English.
- Sentiment of sarcastic comments and emoticon recognition is not considered.

Project Assumptions

In order to use this application, the user must have an Internet connection and a web browser (preferably Google Chrome).Application is developed in English language.

1.4 Organization Of Project Report

The project documentation begins with a brief idea about the software being developed, the significance of the software, and product scope. The second chapter highlights the Software Requirements Specification for the system. It includes the overall product description along ith Functional and Non-Functional Requirements. It also covers the topics related to the planning and management of the project. The third chapter involves the design description of the project. It includes UML Diagrams, Architectural and component-level design. The fourth chapter involves the technical aspect of the project, it includes operational details as well as the preliminary User Interface Design have been provided. The fifth chapter deals with project testing. The sixth chapter covers the results and discussion of the project.An inference based on this result will also be discussed.

Chapter-2

Project Planning and Management

2.1 System Requirement Specification (SRS)

2.1.1 Purpose

This document gives a detailed description of the requirements for Box Office revenue prediction using dual sentiment analysis system. It will also explain system constraints, interface and interactions with other external applications. All the external interface requirements and other non-functional requirements will be properly stated in SRS. This document is primarily intended to be provided to the customer so that he can check if all his requirements are satisfied or not. After his approval, the development team can proceed towards developing the first version of the product.

2.1.2 Document Conventions

The format of this SRS is simple. Bold face and indentation is used on general topics and or specific points of interest. The remainder of the document will be written using the standard font, New Times Roman.

2.1.3 Intended Audience and Reading Suggestions

This document is meant for the project developers, Internal and External project guide, and for the end users of the site. Almost all the chapters will be needed by the system programmers. Chapter 1 and 2 give a detailed idea about what is expected from the project, so it will help developers in implementing the system. Chapter 3 and 4 will interest the user as he will come to know what are system components, how is he supposed to interact with the system ad what is the desired output.

2.1.4 Product Scope

Presently there are systems that mine through this data to predict box office revenues of upcoming movies. There are systems that use the pre-release buzz and polarity of sentiments to figure out how successful the movie will be. But those systems follow a very straightforward technique of sentiment analysis which sometimes leads to wrong interpretation of users sentiments. There are also systems that take number of searches as the parameter, example, more the number of searches, more successful the movie will be. And also, these systems dont consider the additional information, such as, star cast, status of the last few movies of the actors, holiday effect, competitive movies, number of the times the tweet has been retweeted or favorited, etc. These all factors when taken into account, can lead to higher quality of output. Hence, we propose a system that will perform multivariate regression considering all the above stated parameters. Moreover, we will use the technique of Dual Sentiment Analysis on the Twitter data. We will reverse each tweet and considering both things, i.e, how positive the original tweet is and how negative the reversed tweet is or how negative the original tweet is and how positive the reversed tweet is, the final decision will be made if the tweet falls in positive, negative or neutral category. Our system will show a list of the upcoming movies on the home page. When a user will click on any movie of his choice, he will be shown what is the approximate revenue predicted by our system. Along with this, he will also be given the list of all the factors that were considered for the prediction.

2.1.5 Product Perspective

This web application will be accessed through web browser. The user interface will be very user friendly. The user will be required to just click on a movie name in which he is interested. The underlying database will be searched for that movie and all the information, including the revenue predicted and the other static factors will be shown to the user. Our System will also store the predicted revenue of the movie once it is out from theatres, so that we can feedback that data to make more accurate predictions the next time.

Product Functions

1. The application will provide the facility of getting information regarding the selected movie, along with its predicted revenue.
2. The system will also show the sentiment of tweets.

Operating Environment

In order to use this application, the user must have an Internet connection and a web browser (preferably Google Chrome).

Design and Implementation Constraints

Following are some of the constraints to be kept in mind:

1. Due to Twitter API limitations, only a small part of total tweets available can be captured.
2. For more accurate predictions, more time needs to be given to the system.

3. Tweets in English only can be used, as the system only supports English language.
4. Sentiment of sarcastic comments and emoticons in tweets cannot be identified.
5. Only prediction of Bollywood movies is supported.

Assumptions and Dependencies

1. Internet Connection
2. Web Browser.

2.1.6 External Interface Requirements

User Interfaces

The application provides a user friendly Graphical User Interface.
There are two web pages that user(audience) is provided with:

1. Welcome Page:
   The system will provide user a list of upcoming movies. User needs to click on a movie poster to get the prediction.
2. Result page:
   This page gives the revenue predicted of the selected movie along with factors contributing to the result and other movie details like its official trailler.

There are 3 web pages provided for to the Admin of the system:

1. Options Page:
   This page provides the Admin with an option to either reschedule the tweet collection a movie or add a new movie to the database.

2. Rescheduling Page:
   Here, the system provides the admin a facility to start tweet collection of a previously added movie. The admin needs to provide the movie name and hashtag. and click on data collection button. He then needs to start the processing of captured tweets after a certain duration.

3. Add a new movie:
   Here, the admin needs to provide all the movie details specified on the web page like movie name etc and click add movie to add it to the database. He/She should start the tweet collection of the newly added movie and after a certain duration he should start the processing of captured tweets.

Hardware Interfaces

1. Need of a normal commercial server for hosting web application.

Software Interfaces

1. Firefox or google chrome browser is essential to open the website.

2. Eclipse will be needed for the development portion of the project.

3. Java SDK will be needed to implement the application.

4. MySql required for storing the twitter data and also the predicted revenues.

5. Twitter streaming and search Rest APIs are required.

6. OAuth will be used for Authentication.

2.1.7 System Features

These are the functional requirements of the system:

Web Browser and UI features

Our system will be based on Client-server (MVC) architecture. A powerful commodity server is employed. The server will be responsible for accepting client requests and giving them results accordingly. We will also employ a database server which will store all the databases, like, Tweets database.

Description and Priority:

This feature will support multiple clients using the system at the same time.

Stimulus/Response Sequences:

Application requires user initiation.
The user on opening the first page, is provided with a list of upcoming movies.

1. Fetch the prediction result.

If the user is admin, he/she needs to login with correct username and password. He/She is then provided with the following options:

1. Add a new movie.
   (a) Add movie to database.
   (b) Start tweet collection of respective movie.
   (c) Start processing of captured tweets.
2. Reschedule a movie.
   (a) Start tweet collection of respective movie.
   (b) Start processing of captured tweets.

Dual Sentiment Analysis (DSA)

Description and Priority:

DSA is an efficient model to address the polarity shift problem in sentiment classification.Here, not only is the original tweet considered for calculating its sentiment, but also its reverse, and use sentiment scores of both the original and reversed tweet to figure out the final sentiment.

Stimulus/Response Sequences:

DSA will be an internal algorithm used in our system. The user will have no role in triggering this action. This algorithm is triggered whenever revenue is to be predicted.

Regression

Description and Priority:

Regression is the technique which finds out the relation among various variables.

Stimulus/Response Sequences:

Regression is an internal algorithm used in the system. This technique is used to find out the effect of various factors such as start-cast, genre, holiday effect, polarity , etc., on the overall success of the movie.

2.1.8 Other Nonfunctional Requirements

Performance Requirements:

The following should apply:

1. Network connectivity can vary. So, the system should efficiently operate even when there is low connectivity.

2. To reduce the system lag, the processing and calculations are performed beforehand and the results are stored in the database. So that when user wants the data, it just becomes a fetch operation from database.

3. Memory requirements should also be low.

Safety Requirements:

The system will not perform any kernel manipulation operations. It will not degrade, harm or interfere any other running application. The application will be completely safe.

2.1.9 Other Requirements

Security Requirements

There is no involvement of users personal data in our system. Therefore there will be no such security requirements. The only security requirement is that data will be collected through a Twitter account. So, account should not be compromised. That care will be taken by OAuth.

Response Time:

The system shall respond to any request in few seconds from the time of the request submittal.

Software Quality Attributes

Usability:

1. The system should be user friendly and self-explanatory.
2. Since all users are familiar with the general usage of web applications and websites, no specific training should be required to operate the application.

Apart from this, the system should be highly Reliable, Flexible, Robust and easily Testable.

2.2 Project Process Modeling

For development of the system, Incremental Process Model is used. The system is decomposed into a number of components like Data collection module, Data cleaning module, Polarity analyzer module, Reversal module , Regression module and UI Implementation. Each of these components will be designed and built separately. Each component will be delivered when it is completed. This allows partial utilization of product and avoids a long development time. It also avoids long wait. This model of development also helps ease the traumatic eect of introducing completely new system all at once.

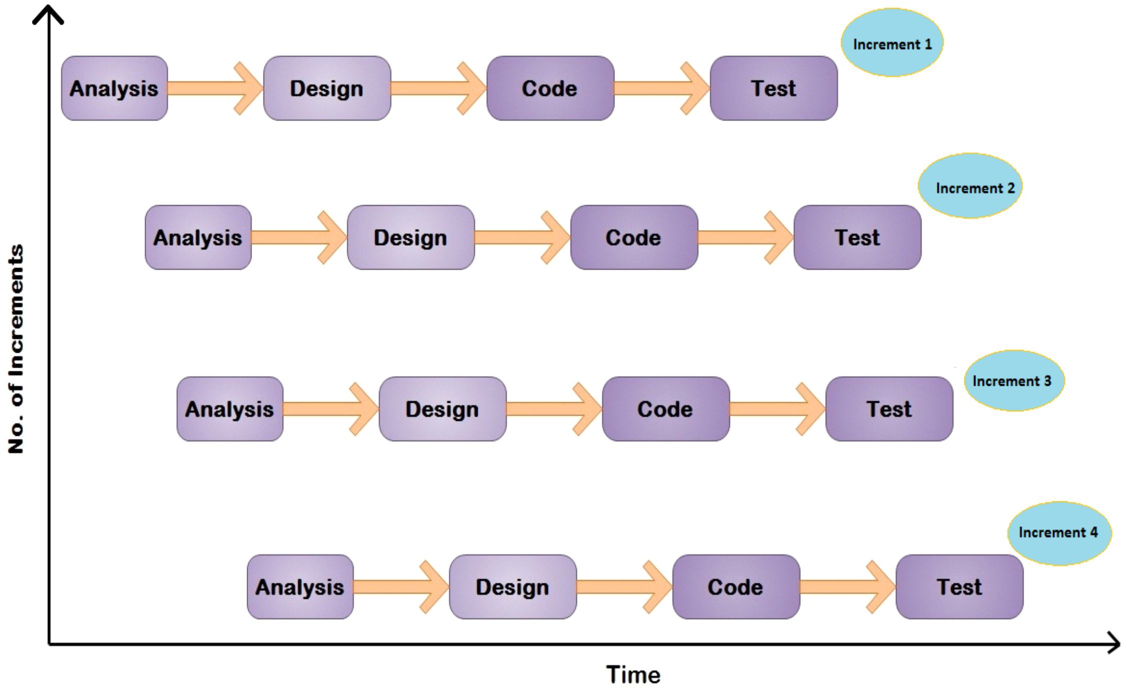


Figure 2.1: System Design Overview

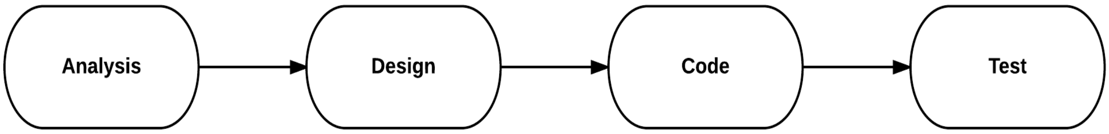


Figure 2.2: System Design Overview

The work breakdown structure of the project is as follows:

- Initiate the project
  - Communicate with the company and requirement gathering
  - Literature survey
  - Define scope
  - Develop SRS
- Plan the project
  - Design mathematical model
  - Design UML and other diagrams
  - Develop test plan
  - Develop risk management plan
  - Feasibility analysis
- Execute the project
- Test the project

2.3 Cost & Efforts Estimates

Table 2.1: Computing Function Point Table

Information domain value	Count		Simple	Average	Complex		
External Inputs	8	×	3	4	**6**	=	48
External outputs	3	×	4	5	**7**	=	21
External Inquiries	1	×	3	**4**	6	=	4
Internal logical files	4	×	7	10	**15**	=	60
External Interface Files	3	×	5	7	**10**	=	30
Total Count							163

Function Point = Total Count $\times$ $[0.65 + 0.01 \times \sum(F_i)]$
= 163 $\times$ $[0.65 + 0.01 \times 44]$
= 177.67

Project Duration = 10 Months
Team Size = 2
Total Effort by 2 people in 10 months = 20 person months.

2.4 Project Scheduling

The estimated Timeline Chart for the entire project is as shown:

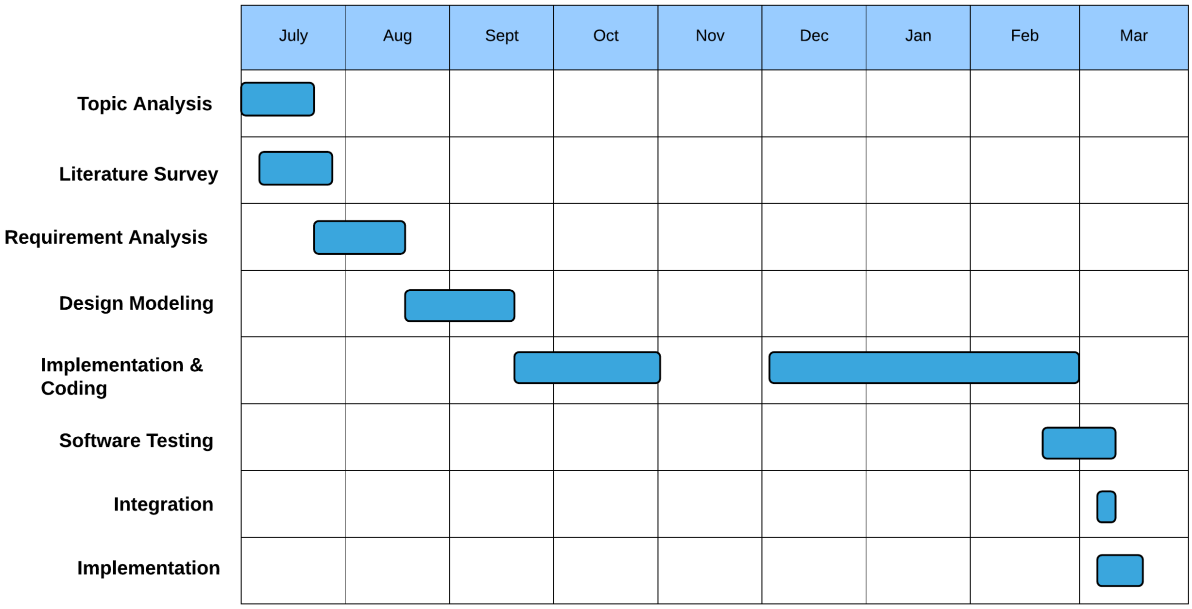


Figure 2.3: Timeline Chart

The following set of activities for the project have been identified :

- Topic Analysis
- Literature Survey
- Requirement Analysis
- Design Modeling
- Implementation and Coding
- Software Testing

- Integration
- Implementation

Chapter-3

Analysis & Design

3.1 System Architecture Diagram

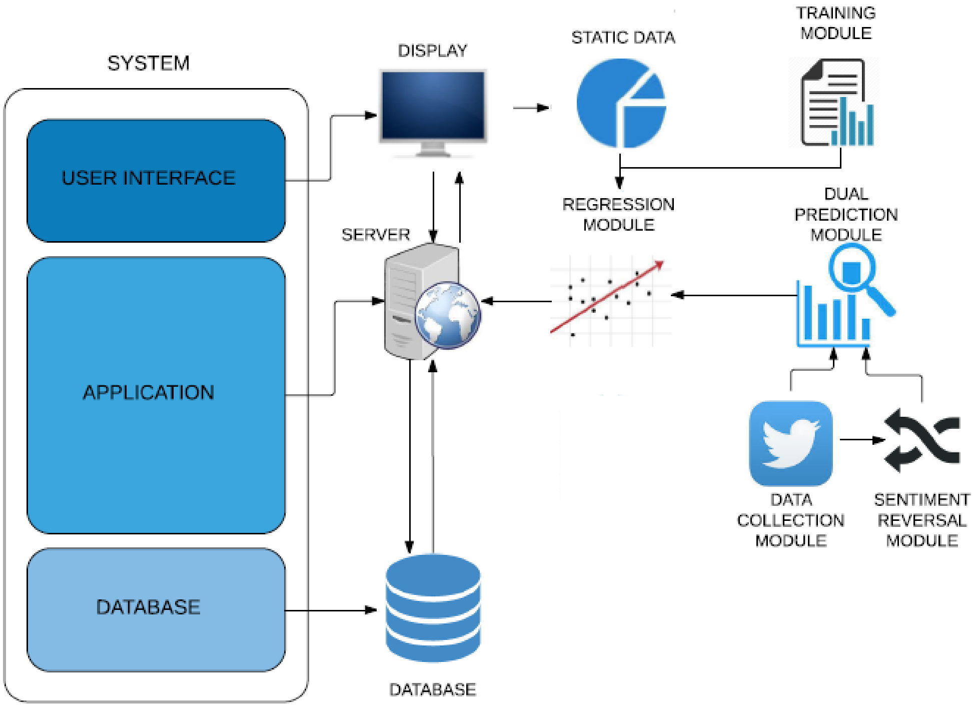


Figure 3.1: Block Diagram

3.2 Use Case Diagram

A Use Case diagram is a type of behavioral diagram defined by the UML created from a use case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goal represented as use case and any dependencies between those use cases. It is a type of diagram that shows a set of use cases, actors and their relationships. It should have a distinct name. It commonly contains

- Use Cases.
- Actors (Primary and Secondary).
- Dependency, Generalization and association relationships.

Below are the Use case diagrams of Box office prediction application.

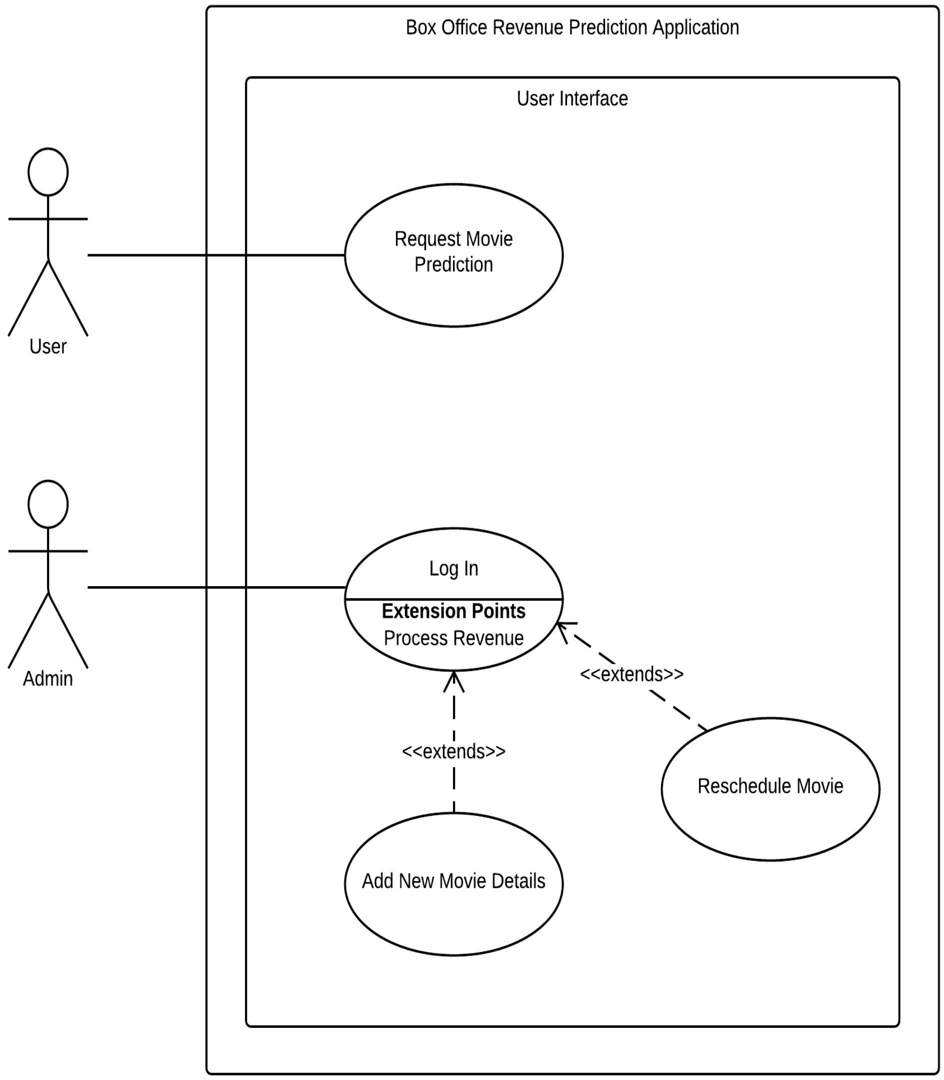


Figure 3.2: Use Case Diagram to view prediction and login.

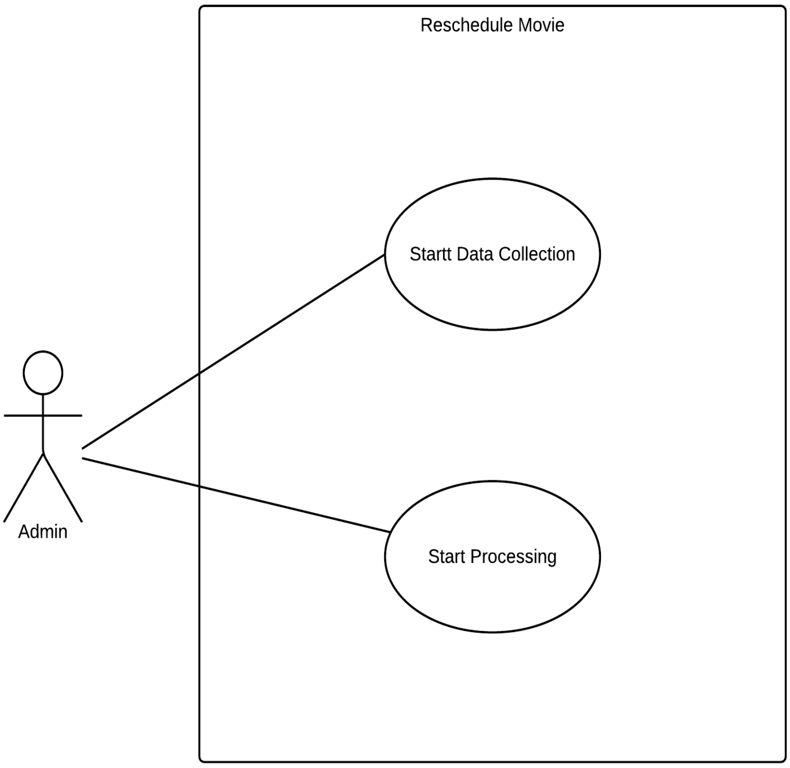


Figure 3.3: Use Case Diagram for rescheduling movie.

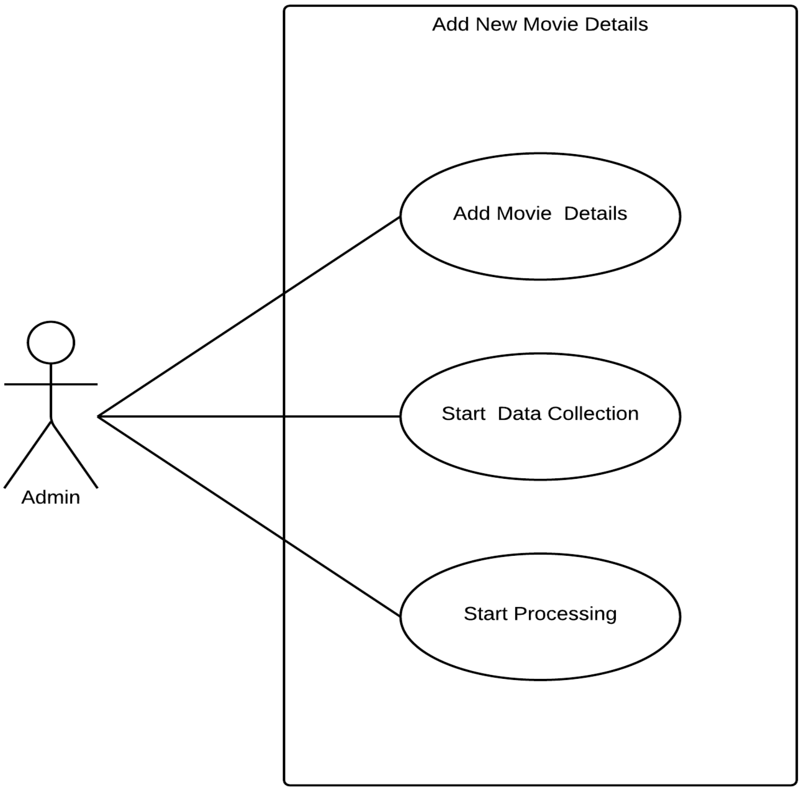


Figure 3.4: Use Case Diagram for adding new movie.

3.3 Class Diagrams

Below given figure represents class diagram of box office prediction application.

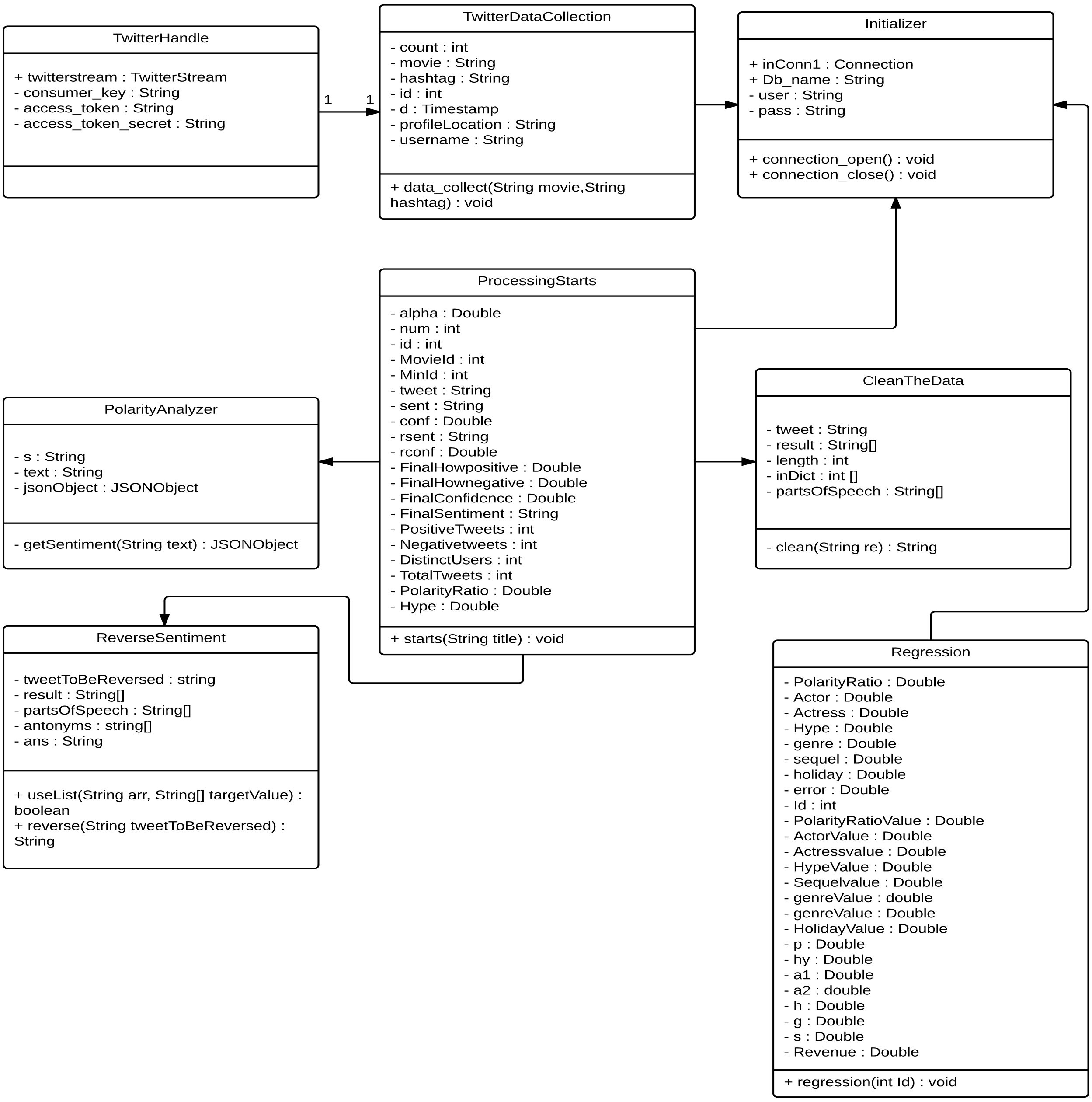


Figure 3.5: Class Diagram

3.4 State Diagrams

Below given are State Chart Diagrams.

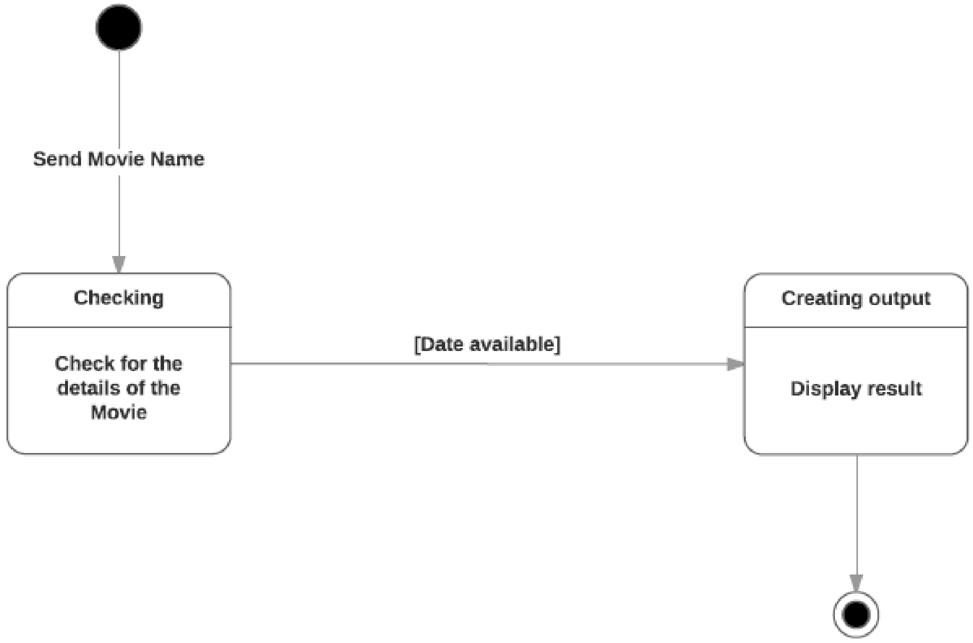


Figure 3.6: State Diagram 1

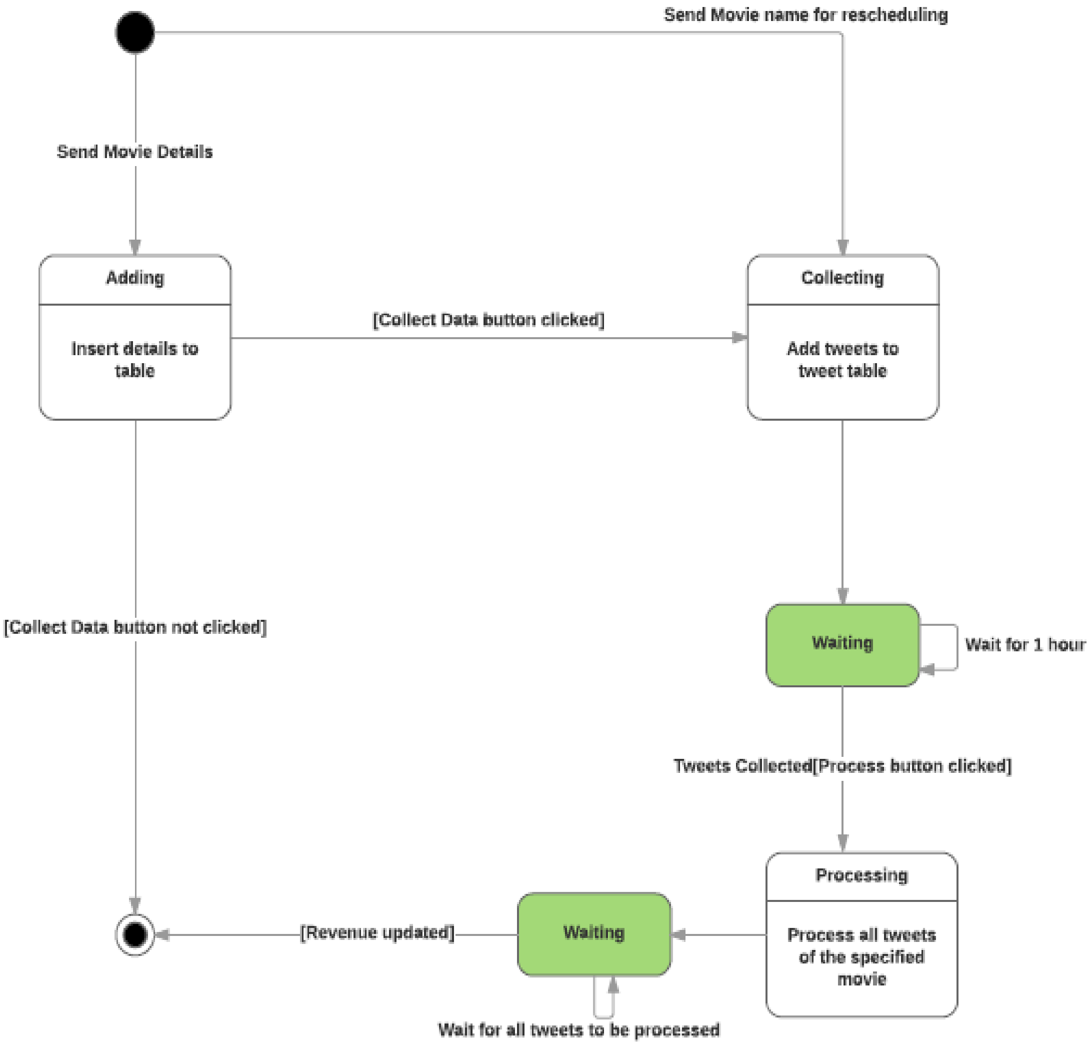

Send Movie name for rescheduling
Send Movie Details
Adding
Insert details to table
[Collect Data button clicked]
Collecting
Add tweets to tweet table
[Collect Data button not clicked]
Waiting
Wait for 1 hour
Tweets Collected[Process button clicked]
Processing
Process all tweets of the specified movie
Waiting
[Revenue updated]
Wait for all tweets to be processed

Figure 3.7: State Diagram 2

3.5 Deployment Diagram

Below given is Deployment Diagram.

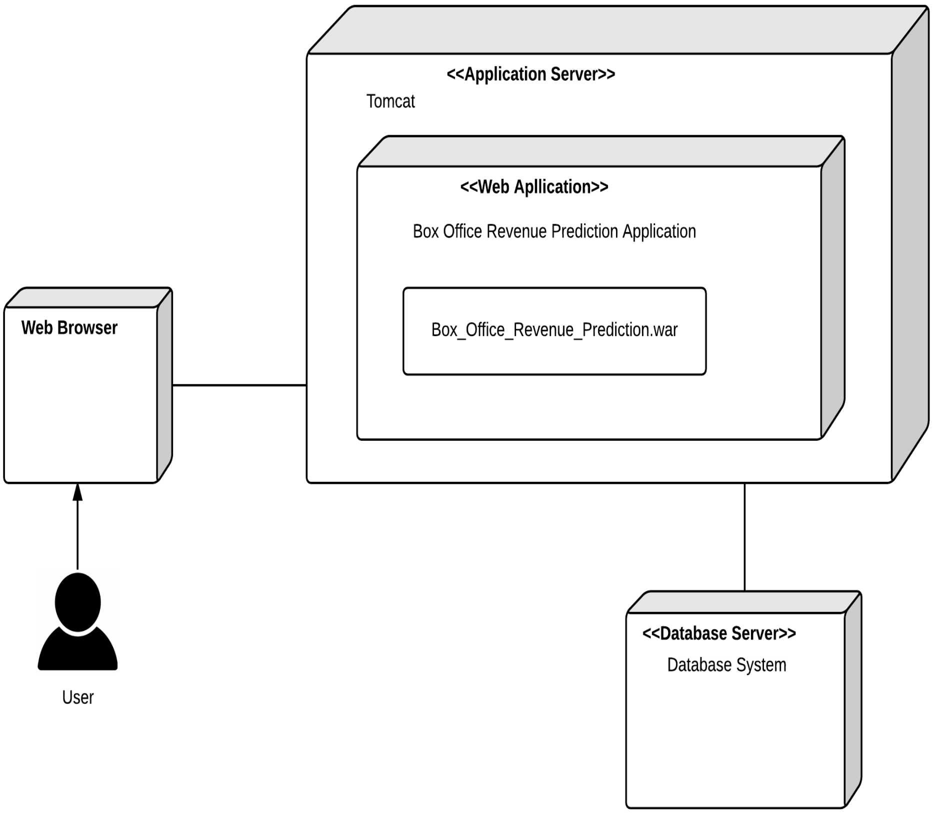


Figure 3.8: Deployment Diagram

3.6 Sequence Diagrams

Below given is Sequence Diagram.

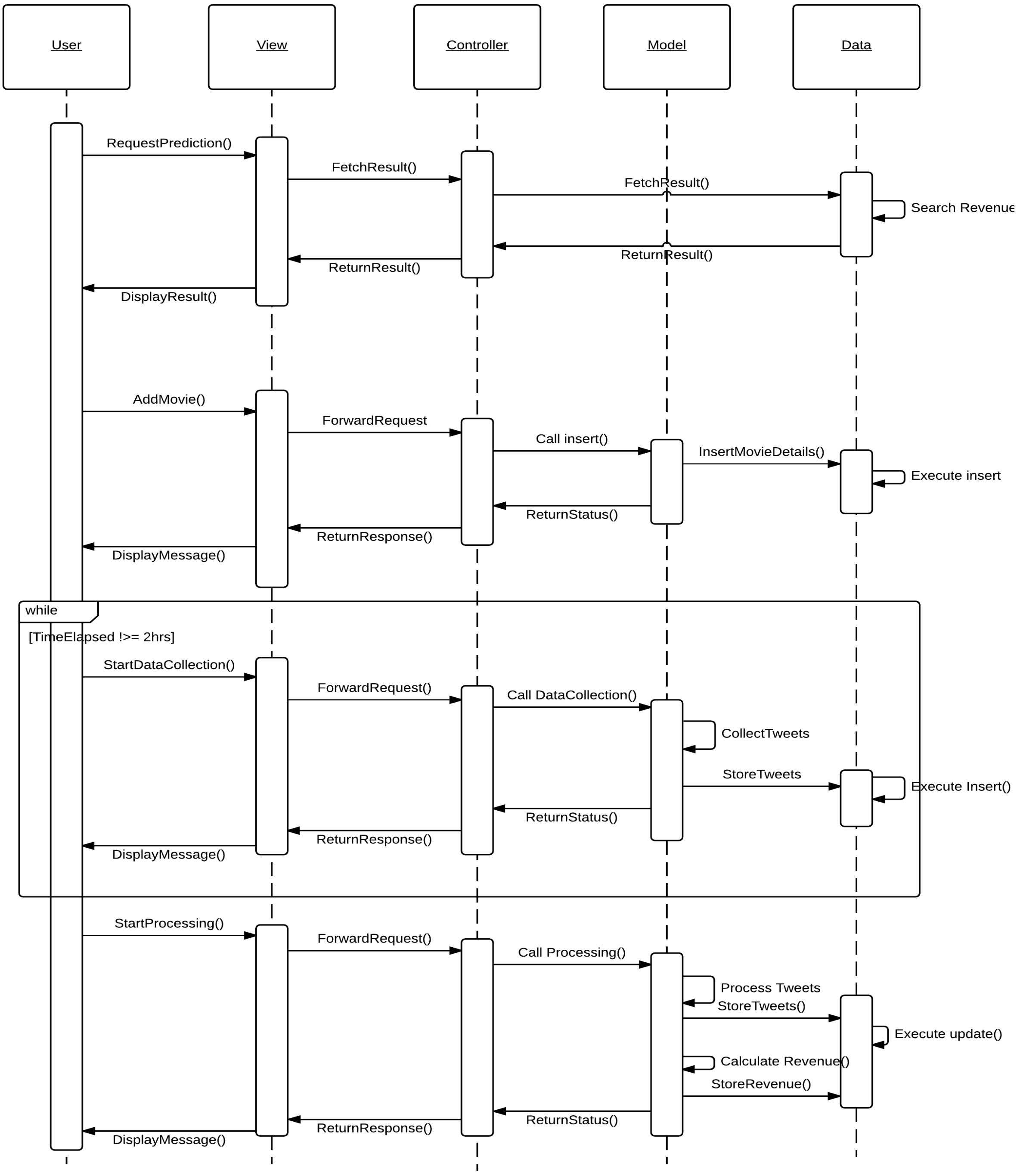


Figure 3.9: Sequence Diagram

3.7 ER Diagrams

Below given is ER Diagram.

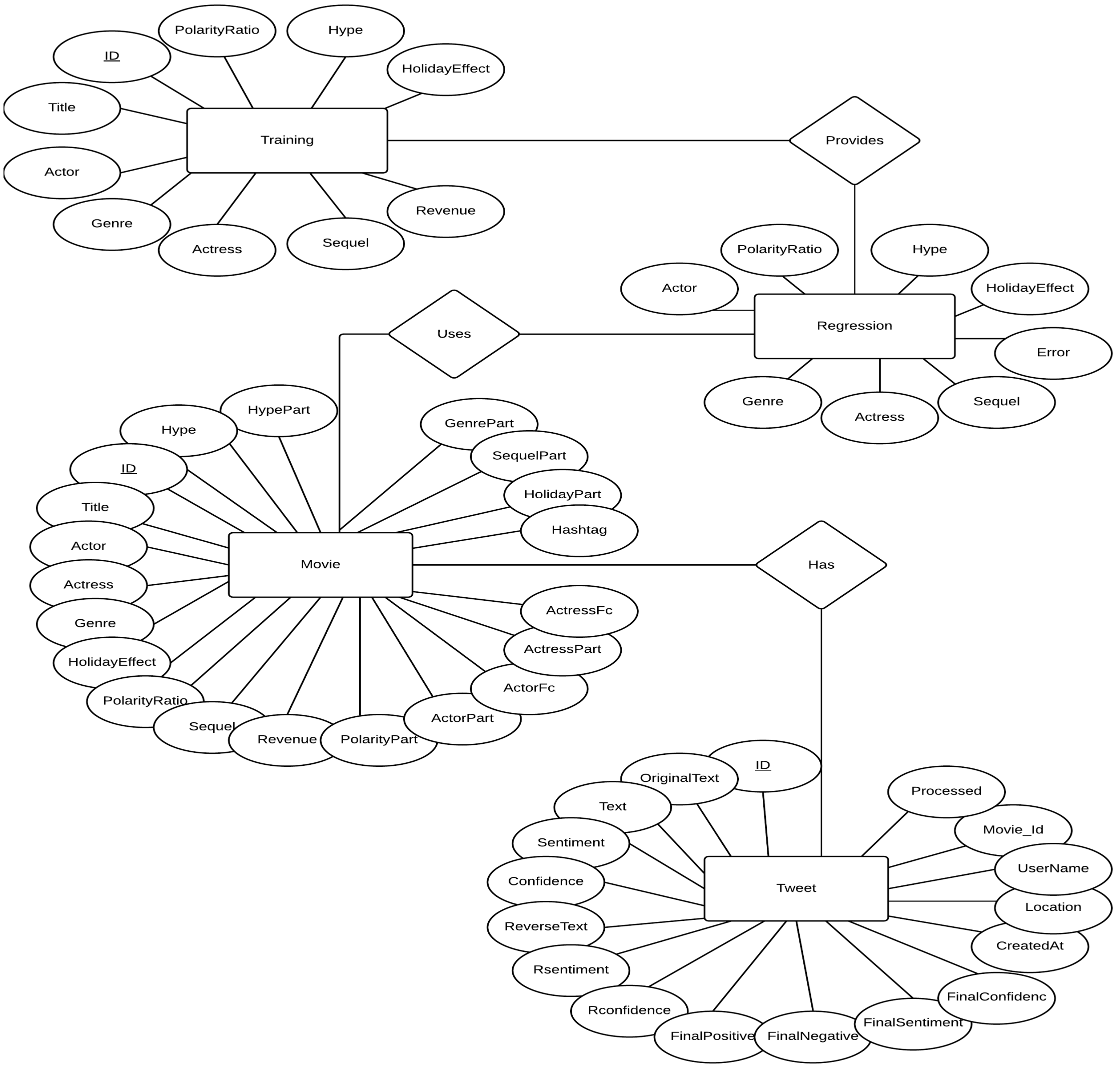


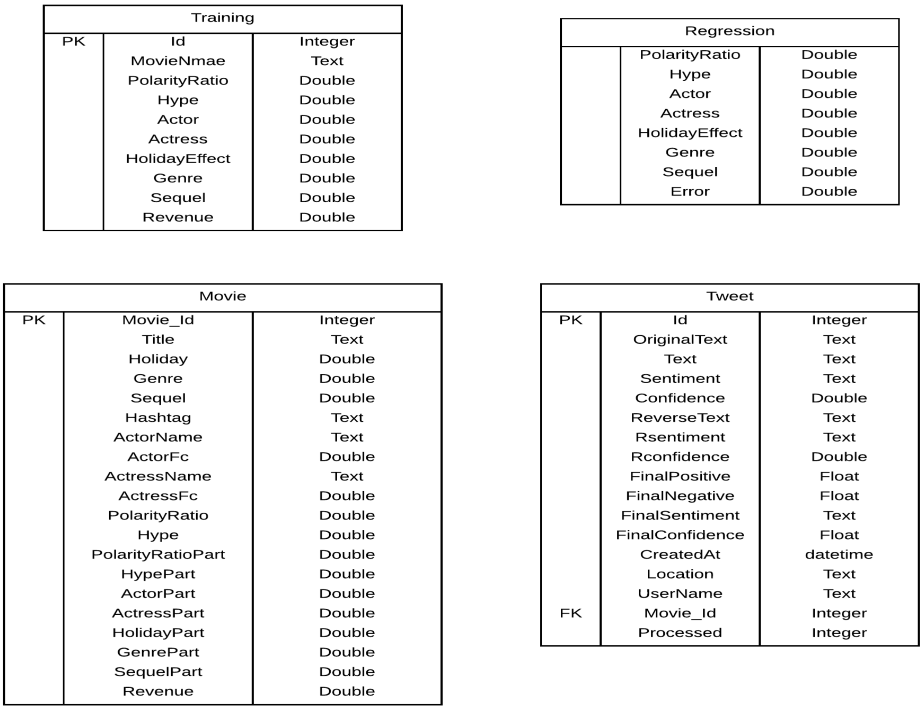
Figure 3.10: ER Diagram

Chapter-4

Implementation & Coding

4.1 Database Schema

4.1.1 Database



Training		
PK	Id	Integer
	MovieNmae	Text
	PolarityRatio	Double
	Hype	Double
	Actor	Double
	Actress	Double
	HolidayEffect	Double
	Genre	Double
	Sequel	Double
	Revenue	Double

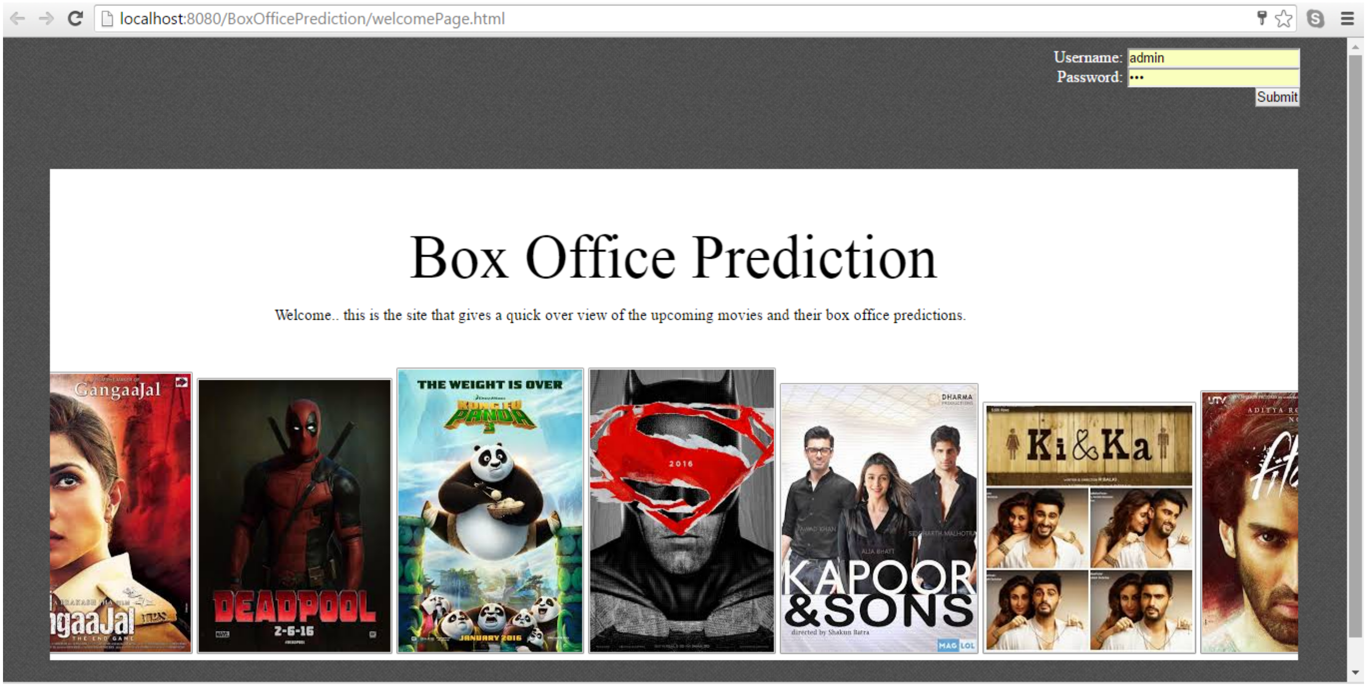
Regression		
	PolarityRatio	Double
	Hype	Double
	Actor	Double
	Actress	Double
	HolidayEffect	Double
	Genre	Double
	Sequel	Double
	Error	Double

Movie		
PK	Movie_Id	Integer
	Title	Text
	Holiday	Double
	Genre	Double
	Sequel	Double
	Hashtag	Text
	ActorName	Text
	ActorFc	Double
	ActressName	Text
	ActressFc	Double
	PolarityRatio	Double
	Hype	Double
	PolarityRatioPart	Double
	HypePart	Double
	ActorPart	Double
	ActressPart	Double
	HolidayPart	Double
	GenrePart	Double
	SequelPart	Double
	Revenue	Double

Tweet		
PK	Id	Integer
	OriginalText	Text
	Text	Text
	Sentiment	Text
	Confidence	Double
	ReverseText	Text
	Rsentiment	Text
	Rconfidence	Double
	FinalPositive	Float
	FinalNegative	Float
	FinalSentiment	Text
	FinalConfidence	Float
	CreatedAt	datetime
	Location	Text
	UserName	Text
FK	Movie_Id	Integer
	Processed	Integer

Figure 4.1: Database Design

4.2 GUI Design



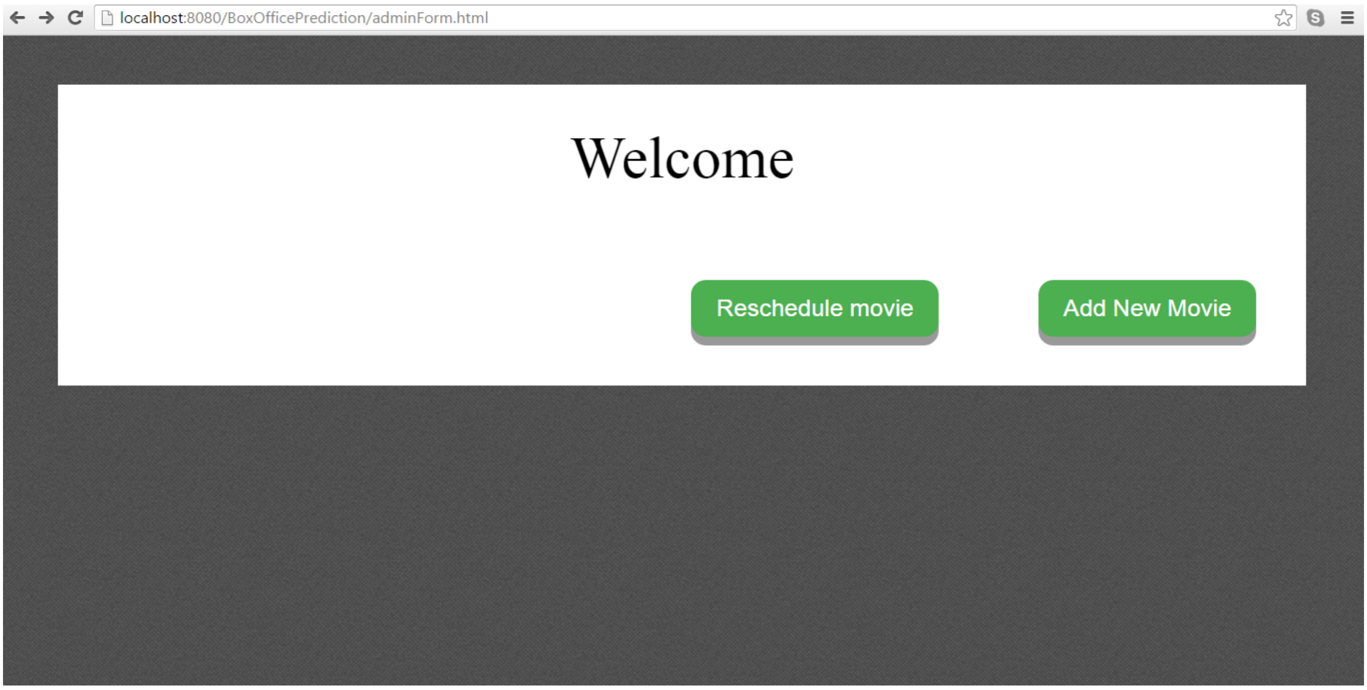
localhost:8080/BoxOfficePrediction/welcomePage.html

Username: admin
Password: •••
Submit

Box Office Prediction

Welcome.. this is the site that gives a quick over view of the upcoming movies and their box office predictions.

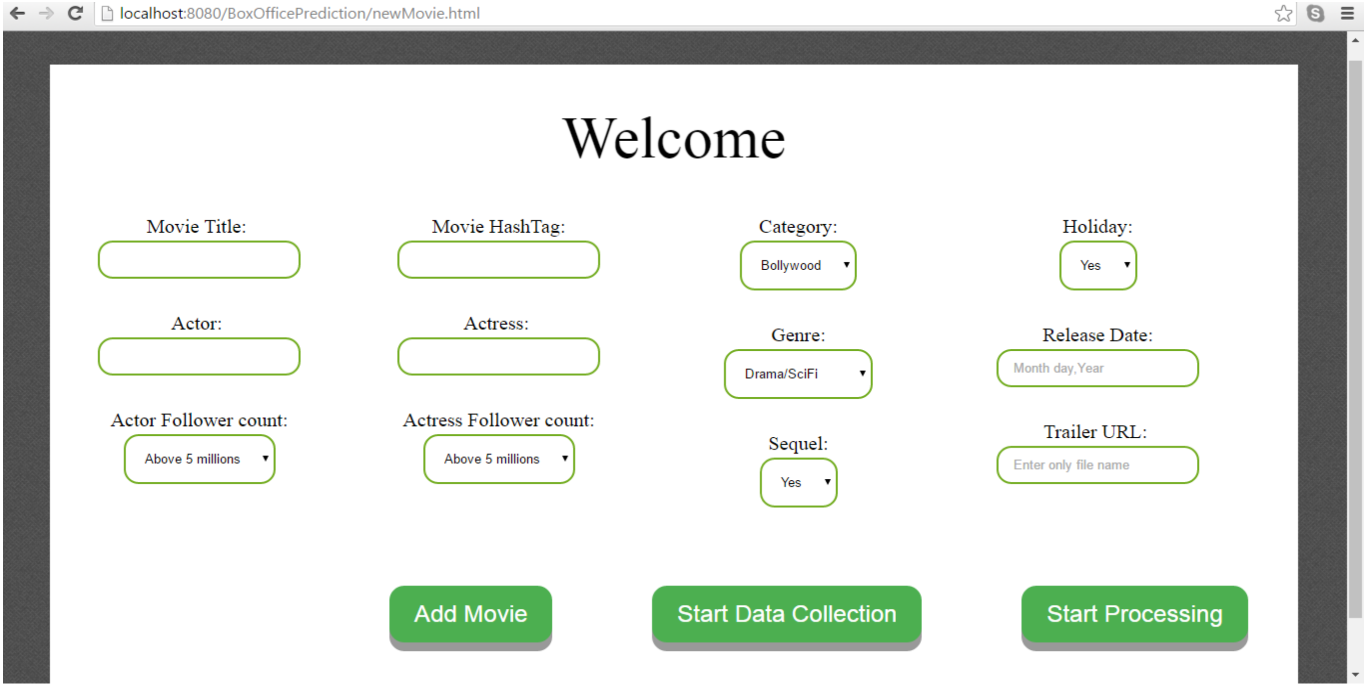
Figure 4.2: Home page



localhost:8080/BoxOfficePrediction/adminForm.html

Welcome

Figure 4.3: Admin Page


localhost:8080/BoxOfficePrediction/newMovie.html

Welcome

Movie Title:

Movie HashTag:

Category: Bollywood

Holiday: Yes

Actor:

Actress:

Genre: Drama/SciFi

Release Date: Month day,Year

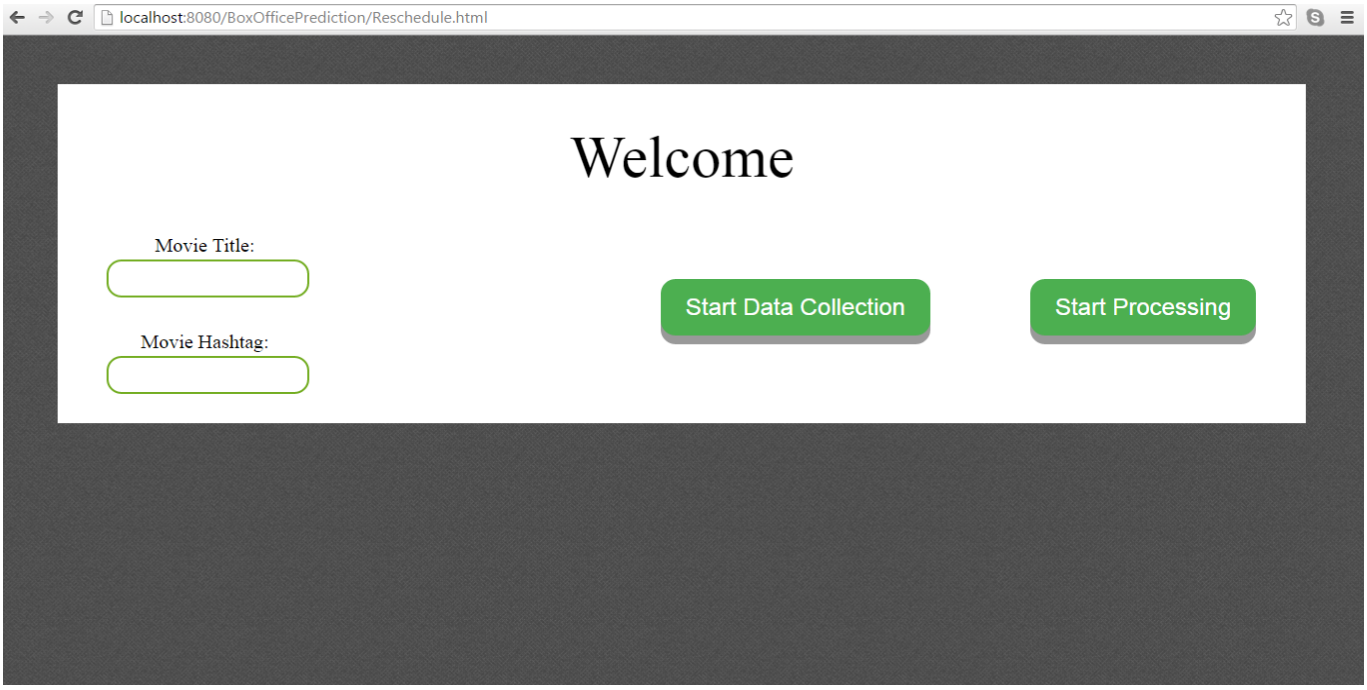
Actor Follower count: Above 5 millions

Actress Follower count: Above 5 millions

Sequel: Yes

Trailer URL: Enter only file name

Figure 4.4: Add New Movie page

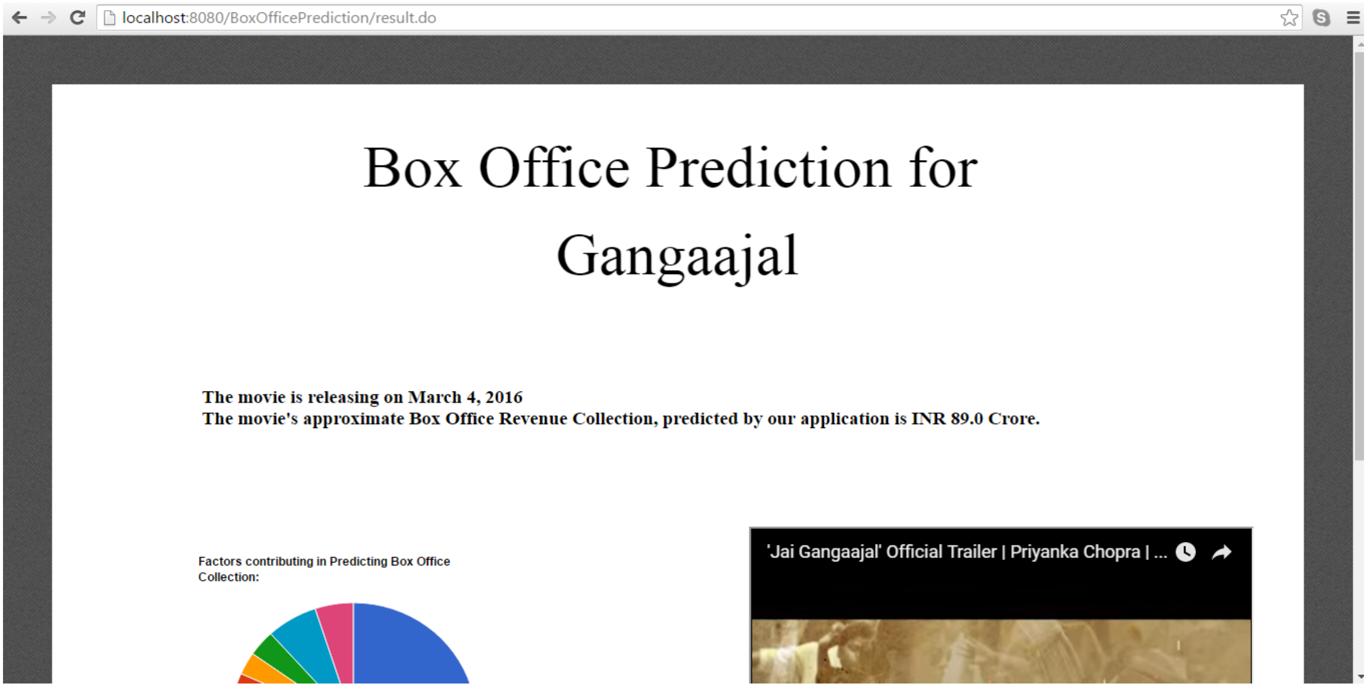

localhost:8080/BoxOfficePrediction/Reschedule.html

Welcome

Movie Title:

Movie Hashtag:

Figure 4.5: Reschedule Movie Page



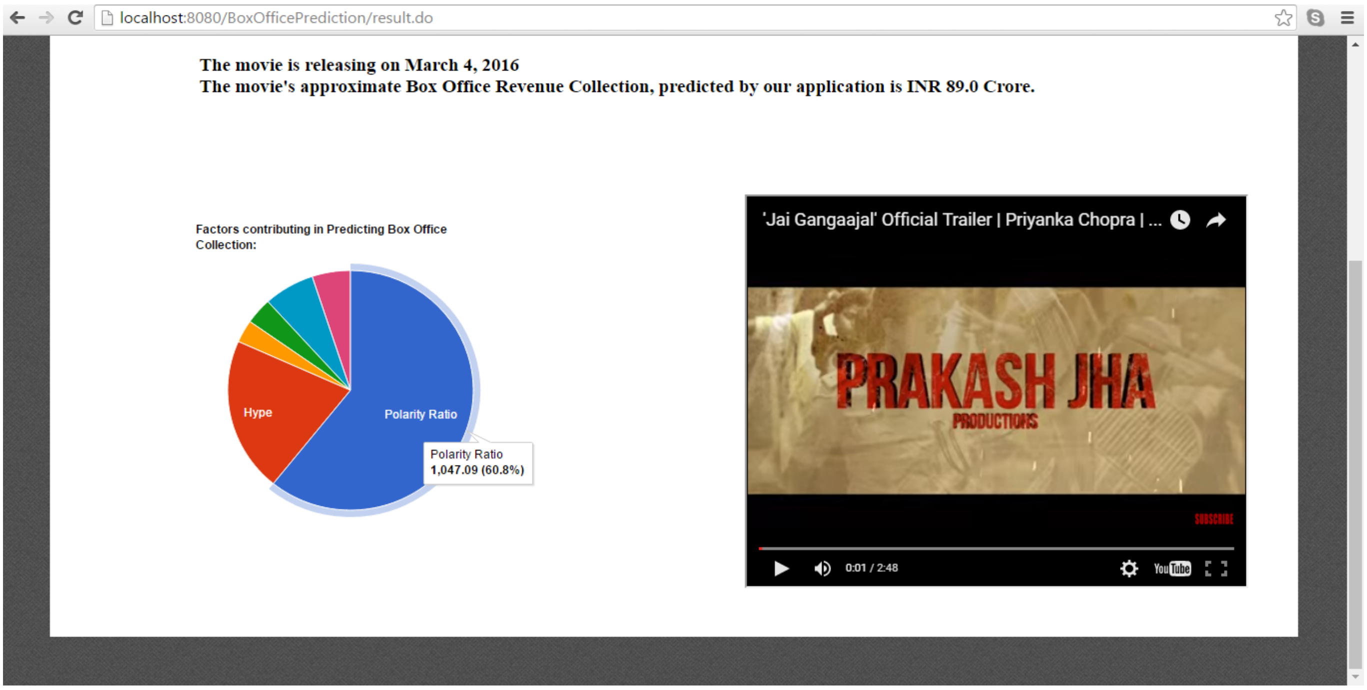
localhost:8080/BoxOfficePrediction/result.do

Box Office Prediction for Gangaajal

The movie is releasing on March 4, 2016
The movie's approximate Box Office Revenue Collection, predicted by our application is INR 89.0 Crore.

Factors contributing in Predicting Box Office Collection:
'Jai Gangaajal' Official Trailer | Priyanka Chopra | ...

Figure 4.6: Result Page



localhost:8080/BoxOfficePrediction/result.do

The movie is releasing on March 4, 2016
The movie's approximate Box Office Revenue Collection, predicted by our application is INR 89.0 Crore.

Factors contributing in Predicting Box Office Collection:
Hype
Polarity Ratio
Polarity Ratio
1,047.09 (60.8%)
'Jai Gangaajal' Official Trailer | Priyanka Chopra | ...
PRAKASH JHA
PRODUCTIONS
SUBSCRIBE
0:01 / 2:48
YouTube

Figure 4.7: Result Page

4.3 Algorithms

4.3.1 Data Collection Module

- Admin has the control to start the Data collection module through User Interface.
- Module runs for given amount of time.
- Module stores the data in database.

```
1 While timeElapsed is greater than given certain amount of time.
1.1 Initialize the Twitter connection.
1.2 Receive Tweets.
1.3 Initialize Database connection.
1.4 Store Tweet text, userName, Location, CreatedAt in database.
1.5 Close database connection.
```

4.3.2 Data Processing Module

- This module is used to Clean the data(tweets), perform Sentiment Analysis and Reverse Sentiment Analysis.
- This module is also triggered by user action through User Interface.

```
1. Initialise Database Connection.
2. Fetch Tweets from database.
3. Clean the data.
4. Calculate Sentiment and Confidence of Original text.
5. Reverse the Original Text.
6. Calculate Sentiment and Confidence of Reverse Text.
7. Calculate the Dual Prediction Score.
8. Calculate Polarity Ratio.
9. Calculate Hype.
10. Store the above results in the database.
```

4.3.3 Regression Module

- This module is used to calculate Final Revenue of the movie.
- This module is called after processing all the tweets stored in the database and gets updated everytime new tweets are added in the database.

```
1. Initialise Database Connection.
2. Fetch training regression coefficients from database.
3. Fetch movie data from database.
4. Calculate movie's regression coefficients.
5. Calculate revenue.
6. Store the above result in the database.
```

4.4 Operational Details

The project is divided into number of different modules. The operational details of each module are given below.

4.4.1 Data Collection

This module is responsible for collecting Twitter data for given amount of time.
It collects tweets and stores the them in database along with other details of the tweet such as location, username of user who tweeted, created at etc. It is called by the servlet when a request by admin to start data collection is made. It displays appropriate message after collection of data.
Below is the sample code of twitter data collection module:

```
public class Twitter_data_collection
{

 static int count=0;
 Statement stmt;
 long startTime,stopTime,elapsedTime;
 static Initializer in=new Initializer();
 PreparedStatement pstmt;

 public int data_collect(final String title, String hashtag) throws
 ClassNotFoundException
     {
      in.connection_open();
         startTime=System.currentTimeMillis();
         StatusListener listener = new StatusListener()
         {
```

```
        public void onStatus(Status arg0)
       {

         Status rt= arg0.getRetweetedStatus();
if(rt==null)
{
     java.util.Date d;
     d = arg0.getCreatedAt();
        String username =
        arg0.getUser().getScreenName();
            count++;
                    try
                    {
                        String sql = "SELECT MOVIE_ID FROM movie
                        where TITLE = ?";
                        pstmt = inConn1.prepareStatement(sql);
                      pstmt.setString(1, title);
                      ResultSet rs1=pstmt.executeQuery();
                       rs1.next();
                       int Id=rs1.getInt(1);


                       pstmt = inConn1.prepareStatement("INSERT
                       INTO tweet(OriginalText,CreatedAt,User_name,MOVIE_ID)
                       VALUES (?,?,?,?,?)");
                       pstmt.setString(1, arg0.getText() );
                      pstmt.setTimestamp(2, new
                      java.sql.Timestamp(d.getTime()));
                      pstmt.setString(3, username);
                      pstmt.setInt(4, Id);
                      pstmt.executeUpdate();
                      }
                      catch (SQLException e)
                      {
e.printStackTrace();
                      }
```

```
                        System.out.println("Total number of tweets
                         captured : "+count);

                  }//if of retweet

  stopTime=System.currentTimeMillis();
  elapsedTime=stopTime-startTime;
  if(elapsedTime>120000)
                        {
                        JOptionPane.showMessageDialog(null, "Shutting
                        Down....\n Data collection done!
                        \n You can start Processing now.");
                        twitterStream.shutdown();
                        try
                        {
in.connection_close();
 }
                        catch (SQLException e)
                        {
// TODO Auto-generated catch block
e.printStackTrace();
 }

                       }//if of time
       }//onstatus

        };
        FilterQuery fq = new FilterQuery();
        String keywords[] = {hashtag};
        //which hashtags to follow, movie website

        fq.track(keywords);
        twitterStream.addListener(listener);
        twitterStream.filter(fq);
        return 1;
       }
 }
```

4.4.2 Data Cleaning

This module helps clean the tweet by removing hashtags, special symbols, URLs, extended words etc. It receives tweet from processing and after cleaning returns the corrected tweet back to processing module.
Below is sample code for the same.

```
public class Clean_The_Data
{
private final static String REGEX = "((www\\.[\\s]+)|(https?://[^\\s]+))";
private final static String STARTS_WITH_NUMBER = "[1-9]\\s*(\\w*)";
static RiWordnet wordnet = new RiWordnet(null);
static Stemmer obj=new Stemmer();

String clean(String re,String actor,String actress,String moviename)
throws ParseException
 {
 String tweet="";
 String actorregex = actor.trim().replaceAll(" +", "");
 //remove multiple spaces

 String actressregex = actress.trim().replaceAll(" +", "");
 //remove multiple spaces

 String movieregex = moviename.trim().replaceAll(" +", "");
 //remove multiple spaces

/*********************HASH TAGS AND URLS*********************/

    re = re.replaceAll(REGEX, ""); //remove url
 re=re.replaceAll("[-+.^:,!?()/[/]]",""); //remove special characters
 re = re.replaceAll("-", " "); //remove -
 re= re.replaceAll(STARTS_WITH_NUMBER, "");
 re = re.trim().replaceAll(" +", " "); //remove multiple spaces
```

```
/*******************PREPOSITIONS*******************/

List<String> list = new ArrayList<String>();
list.addAll(Arrays.asList(actor,actress,moviename,actorregex,
        actressregex,movieregex,"you","i","this","that","of","for","to","the","with"
        "where","there","here","from","my","your","her","and","his"));

String[] result = re.split("\\s");
int length=result.length;
int[] inDict=new int[length];
for(int k =0;k<length; k++)
{
inDict[k]=1;
}
//initializing inDict

for(int k =0;k<length; k++)
{
//System.out.println(result[k]);
if(result.length>1 &&(result[k].charAt(0)==’#’ ||
result[k].charAt(0)==’@’))
{
 result[k]=result[k].substring(1);

}
else
{
if(list.contains(result[k]))
 continue;
 else
 {
 String[] partsofspeech =
 wordnet.getPos(result[k]); //updating inDict
 if(partsofspeech.length==0)
 {
  inDict[k]=0;
  }
  }
 }//else

 }//for of hash

 /*******************EXTENDED WORDS********************/
 for(int k =0;k<length; k++)
 {
 if(inDict[k]==0)
 {

 int len=result[k].length();
 for(int i=len-1;i>=1;i--)
  {
  if(result[k].charAt(i)==result[k].charAt(i-1)) //repeated
  {
String[] partsofspeech =
wordnet.getPos(result[k]);
if(partsofspeech.length==0)      //not in dictionary
{
result[k]=result[k].substring(0,i)+result[k].substring(i+1);
len=result[k].length();
String[] partsofspeech1
= wordnet.getPos(result[k]);
if(partsofspeech1.length>0)
{
inDict[k]=1;
break;
}
}


  }// outer if
  }// inner for
 }//end of indict if
 }// outer for


 /****************************STEMMING***************************/

   tweet = Arrays.toString(result);
   tweet=obj.StemText(tweet,inDict);
   tweet=tweet.substring(1,tweet.length()-1);
   return tweet;

}//main
}
```

4.4.3 Sentiment Analysis and Reverse Sentiment

This module is called by processing module that processes tweets. After the tweets are cleaned , they are sent to Sentiment Analysis API one by one. The API returns the sentiment of the tweet as well as confidence. It further calculates the reverse sentiment by calling the reverse sentiment module. It constructs a reverse tweet with the help of WordNet. The reverse tweet is sent to Sentiment Analyser which returns the sentiment and confidence for reverse tweet. The sentiment and confidence of original as well as the reverse tweet are then stored in the database. Below sample shows the sample code of the above mentioned module.

```
POLARITY ANALYZER
public class PolarityAnalyzer
{

JSONObject getSentiment(String text) throws
ParseException, UnirestException
    {

     HttpResponse<JsonNode> response = Unirest.post("https://twinword
     -sentiment-analysis.p.mashape.com/analyze/")
     .header("X-Mashape-Key",
      "4wKznV7I02mshGpKv5mXf4rBoGFSp1HABEzjsncV0K975wnLh6")
     .header("Content-Type", "application/x-www-form-urlencoded")
     .header("Accept", "application/json")
     .field("text", text)
     .asJson();

     String s = response.getBody().toString();
     JSONParser parser = new JSONParser();
     Object obj = parser.parse(s);
     JSONObject jsonObject = (JSONObject) obj;
     return jsonObject;
   }


}

REVERSE SENTIMENT
public class Reverse_sentiment
{

public static boolean useList(String[] arr, String targetValue)
{
return Arrays.asList(arr).contains(targetValue);
}

String reverse(String tweettobereversed)
{

tweettobereversed=tweettobereversed.trim().replaceAll(" +", " ");

String[] result = tweettobereversed.split("\\s");
String ans ="";
int flag=0;

int k;
System.setProperty("wordnet.database.dir","C:\\Program Files (x86)
\\WordNet\\2.1\\dict");
AdjectiveSynset adjSynset;
```

```
AdjectiveSynset[] s = null;
WordNetDatabase database = WordNetDatabase.getFileInstance();
RiWordnet wordnet = new RiWordnet(null);




//*******************stem************************

for(k =0;k<result.length; k++)//awesome,movie
{

String[] partsofspeech = wordnet.getPos(result[k]);
if(partsofspeech.length!=0)
{
boolean adj=useList(partsofspeech,"a");
if(!adj)
{
Synset[] nounSynset=database.getSynsets(result[k],SynsetType.NOUN);
if(nounSynset.length==0)
{
Synset[] advSynset=database.getSynsets(result[k],SynsetType.ADVERB);
if(advSynset.length!=0)
{
WordSense[] wadv=((AdverbSynset) advSynset[0]).getPertainyms(result[k]);
result[k]=wadv[0].getWordForm();
}

}
else
{
WordSense[] wn=nounSynset[0].getDerivationallyRelatedForms(result[k]);
if(wn.length!=0)
result[k]=wn[0].getWordForm();
}
}
```

```
}

}

//****************reverse*******************

for(k =0;k<result.length; k++)//awesome,movie
{
if(result[k].length()==1)
{
ans+=result[k]+" ";
continue;
}

try
{
if(result[k-1].equals("not") || result[k-1].equals("don’t"))

{
ans = ans.replace(result[k-1], "");
ans+= result[k]+" ";
continue;
}
}
catch(Exception e)
{

}
Synset[] synsets = database.getSynsets(result[k]);
if(synsets.length!=0)
//word exists in wordnet
{

for(Synset synset : synsets)
//check if there is a direct antonym
{
```

```
WordSense[] ws = synset.getAntonyms(result[k]);
if(ws.length==0)
//no direct, so check if there is indirect antonym
{

SynsetType type = synset.getType();
if (type.equals(SynsetType.ADJECTIVE_SATELLITE))
//if synset of this type
{
flag=1;
adjSynset = (AdjectiveSynset)(synset);
s=adjSynset.getSimilar();
if(s.length!=0)
//check if no indirect antonym
{
    String[] l=s[0].getWordForms();
    WordSense[] wordsenses =
     s[0].getAntonyms(l[0]);
ans+=wordsenses[0].getWordForm()+" ";
break;

}
else
{
ans+=result[k]+" ";
flag=1;
break;
}
}//if type is adj or satellite


}//if no direct antonym
else
{
ans+=ws[0].getWordForm()+" "; //add  direct antonym
flag=1;
break;

}


 }//for each synset

if(flag==0)
ans+=result[k]+" ";

 }//check if word exists in wordnet
else
ans+=result[k]+" ";
flag=0;


}//result

return ans;

}
}
```

4.4.4 Processing Module

This module is responsible for calling the sentiment analysis and reverse sentiment module. It further calculates the dual prediction score based on the original and reverse sentiment of the tweet. It calculates how positive or how negative a tweet is and based on this data, it calculates Polarity Ratio and hype.
Below is sample code of above mentioned module.

```
public class Processing_Starts
{
    static Double alpha=0.6;
static Statement stmt;
static Initializer in=new Initializer();
    static PolarityAnalyzer pol=new PolarityAnalyzer();
    static  Reverse_sentiment rev=new Reverse_sentiment();
    static Clean_The_Data ctd=new Clean_The_Data();

    public void starts(String title)throws
    ClassNotFoundException, SQLException
    {
     in.connection_open();
     //connect to database
     int num=0;
     int id = 0;
     PreparedStatement pstmt;
     ResultSet rs1;

        System.out.println("Processing started\n");
        String SQL1 = "SELECT MOVIE_ID FROM movie where TITLE = ?";
        //get id of the movie from movie name

           pstmt = inConn1.prepareStatement(SQL1);
         pstmt.setString(1, title);
         rs1=pstmt.executeQuery();
            rs1.next();
          int MovieId=rs1.getInt(1);


      pstmt = inConn1.prepareStatement("Select MIN(Id) from tweet where MOVIE_ID=?
      and Processed=?");
          pstmt.setInt(1, MovieId);
          //get id of the tweet from which processing has to be started

      pstmt.setInt(2, 0);
        rs1=pstmt.executeQuery();
        rs1.next();
       int MinId=rs1.getInt(1);


pstmt = inConn1.prepareStatement("Select ActorName,ActressName
from movie where
MOVIE_ID=?");
//get total number of tweets
        pstmt.setInt(1,MovieId);
        rs1=pstmt.executeQuery();
        rs1.next();
        String actor=rs1.getString(1);
    String actress=rs1.getString(2);


    pstmt = inConn1.prepareStatement("create view temp as Select * from
     tweet where MOVIE_ID=?
     and Id>=?");
        pstmt.setInt(1,MovieId);
        pstmt.setInt(2,MinId);
        pstmt.executeUpdate();

        pstmt = inConn1.prepareStatement("SELECT Id,OriginalText FROM temp");
        ResultSet rs = pstmt.executeQuery();

         while(rs.next())
         {
          try
          {

                      String FinalSentiment;
             Double FinalHowPositive=null,FinalHowNegative =
             null,FinalConfidence=null;
             String tweet = rs.getString("OriginalText");
             id = rs.getInt("Id");
             System.out.println("tweet : "+tweet);
          tweet=ctd.clean(tweet,actor,actress,title);
              JSONObject jo = null,rjo = null;
              jo=pol.getSentiment(tweet);
              String sent =(String) jo.get("type");
              System.out.println("Sentiment"+sent);
      Double conf =(Double) jo.get("score");

       conf=Math.abs(conf);
       System.out.println("conf : "+conf);
  String reversed=rev.reverse(tweet);
              rjo=pol.getSentiment(reversed);
  String rsent =(String) rjo.get("type");
  System.out.println("RSentiment"+rsent);
              Double rconf =(Double) rjo.get("score");
       rconf=Math.abs(rconf);
       System.out.println("rconf : "+rconf);




   /****************** DUAL PREDICTION****************/

          if(sent.equals("positive") && rsent.equals("positive"))
        {
          FinalHowPositive = ((1-alpha)*conf)+(alpha*(1-rconf));
          FinalHowNegative = ((1-alpha)*(1-conf))+(alpha*rconf);
        }

        else if(sent.equals("negative") && rsent.equals("negative"))
        {
          FinalHowPositive = ((1-alpha)*(1-conf))+(alpha*rconf);
          FinalHowNegative = ((1-alpha)*conf)+(alpha*(1-rconf));
        }

        else if(sent.equals("positive") && rsent.equals("negative"))
        {
          FinalHowPositive = ((1-alpha)*conf)+(alpha*rconf);
          FinalHowNegative = ((1-alpha)*(1-conf))+(alpha*(1-rconf));
        }

        else if(sent.equals("negative") && rsent.equals("positive"))
        {
          FinalHowPositive = ((1-alpha)*(1-conf))+(alpha*(1-rconf));
          FinalHowNegative = ((1-alpha)*conf)+(alpha*rconf);
        }
       else
        {
       FinalHowPositive=FinalHowNegative=conf;
        }

         System.out.println("FinalHowPositive : "+FinalHowPositive);
        if(FinalHowPositive>FinalHowNegative)
        {
        FinalSentiment="positive";
        FinalConfidence=FinalHowPositive;
        }
       else if(FinalHowPositive==FinalHowNegative)
        {
        FinalSentiment=sent;
       FinalConfidence=conf;
        }
        else
        {
        FinalSentiment="negative";
       FinalConfidence=FinalHowNegative;
        }
        System.out.println(" FinalSentiment : "+
         FinalSentiment);
  /*****************************************************************/

                   pstmt = inConn1.prepareStatement("Update temp
                   SET Text=?,
                   Sentiment=?,
                   Confidence=?,
                   ReverseText=?,
                   Rsentiment=?,
                   Rconfidence=?,
                   FinalHowPositive=?,
                   FinalHowNegative=?,
                   FinalSentiment=?,
                   FinalConfidence=?,
                   Processed=?  where Id=?");

                   pstmt.setString(1, tweet );
                   pstmt.setString(2, sent);
                   pstmt.setDouble(3, conf);
                 pstmt.setString(4, reversed );
                 pstmt.setString(5, rsent);
                 pstmt.setDouble(6, rconf);
                 pstmt.setDouble(7, FinalHowPositive);
                 pstmt.setDouble(8, FinalHowNegative);
                 pstmt.setString(9, FinalSentiment );
                 pstmt.setDouble(10, FinalConfidence);
                   pstmt.setInt(11, 1);
               pstmt.setLong(12, id);
               pstmt.executeUpdate();

                      num++;
                      System.out.println("Number of tweets processed : "+num);

           }//try

         catch(Exception e)
         {
          //e.printStackTrace();
           pstmt = inConn1.prepareStatement("DELETE from temp where ID=?");
           //delete tweets which could not be processed
           stmt = inConn1.createStatement();
           pstmt.setInt(1,id);
           pstmt.executeUpdate();

         }
         }//while
```

```
   int PositiveTweets,NegativeTweets;
double PolarityRatio,Hype,RateOfTweets=0,TotalTweets,DistinctUsers;
 pstmt = inConn1.prepareStatement("Select count(*) from tweet where
 Sentiment=? and MOVIE_ID=?");
 //get number of positive tweets

    pstmt.setString(1,"positive");
    pstmt.setInt(2,MovieId);
     rs1=pstmt.executeQuery();
     rs1.next();
   PositiveTweets=rs1.getInt(1);
     System.out.println("positive tweets : \n"+PositiveTweets);


     pstmt=inConn1.prepareStatement("Select count(*) from tweet where
     Sentiment=? and MOVIE_ID=?");
     //get number of negative tweets

    pstmt.setString(1,"negative");
    pstmt.setInt(2,MovieId);
    rs1=pstmt.executeQuery();
    rs1.next();
     NegativeTweets=rs1.getInt(1);
    System.out.println("negative tweets : \n"+NegativeTweets);

    if(NegativeTweets==0)
     PolarityRatio=100;
    else
     PolarityRatio=PositiveTweets/NegativeTweets;

    System.out.println("ratio :"+PolarityRatio);
    pstmt = inConn1.prepareStatement("Select count(distinct User_name)
    from tweet where MOVIE_ID=?");
    //get number of distinct users who tweeted

    pstmt.setInt(1,MovieId);
    rs1=pstmt.executeQuery();
    rs1.next();
    DistinctUsers=rs1.getDouble(1);

    pstmt = inConn1.prepareStatement("Select count(*) from tweet where
    MOVIE_ID=?");
    //get total number of tweets
    pstmt.setInt(1,MovieId);
    rs1=pstmt.executeQuery();
    rs1.next();
    TotalTweets=rs1.getDouble(1);
    System.out.println("total tweets:" +TotalTweets+" distinct :
    "+DistinctUsers);

    Hype=(double)(DistinctUsers/TotalTweets)+RateOfTweets;
    System.out.println("hype:" +Hype);


    pstmt=inConn1.prepareStatement("Update movie set
    PolarityRatio=?,Hype=? where TITLE=?");
    //update values
    pstmt.setDouble(1,PolarityRatio);
    pstmt.setDouble(2,Hype);
    pstmt.setString(3,title);
    pstmt.executeUpdate();

    pstmt=inConn1.prepareStatement("drop view temp");
     //drop the view
     pstmt.executeUpdate();

     in.connection_close(); //close database connection
```

```
            Regression reg = new Regression(); //call regression
            reg.regression(MovieId);

        }//starts

    }//class
```

Chapter-5

Testing

5.1 Acceptance Testing

Table 5.1: Acceptance Testing

Sr. No.	Acceptance Requirement	Critical		Test Result		Comments
		Yes	No	Accept	Reject	
1.	Application's GUI should be Simple and User-friendly.		✓	✓		This feature helps users to view information more efficiently .
2.	Application should provide Login facility to the Administrator.	✓		✓		This feature is important since all the admin activities are done by admin after Log In.
3.	Application should provide the options such as add new movie details, collect tweets , process tweets and calculate final revenue of the movie to the Administrator.	✓		✓		This feature enables admin to carry out processes necessary for predicting the revenue of requested movie.
4.	Application should provide the prediction along with other details of the movie to the audience.	✓		✓		This feature ensures user satisfaction.
5.	Application should work smoothly	✓		✓		This ensures that the application can be used properly.

5.2 Unit Testing

5.2.1 Data Collection Module

Table 5.2: Unit Testing

Sr. No.	Test Cases	Critical		Test Result		Comments
		Yes	No	Accept	Reject	
1.	Twitter Authentication keys should be provided correctly	✓		✓		Twitter tweets can be captured through Twitter API only after authentication.
2.	Movie name and hashtag should be provided correctly	✓		✓		Since, tweets are captured based on the movie hashtag.
3.	Text part of tweet should be filtered and stored respective to their movie names	✓		✓		This feature is important for further processing.
4.	The module should run smoothly		✓	✓		This feature helps capture tweets efficiently
5.	Module should run for given amount of time.	✓		✓		This is an important feature since data collection needs to be rescheduled.
6.	Twitter connection should be closed after collection period.	✓		✓		The connection should be closed and opened when needed.

5.2.2 Data Processing Module

Table 5.3: Unit Testing

Sr. No.	Test Cases	Critical		Test Result		Comments
		Yes	No	Accept	Reject	
1.	Module should remove URLs from the tweets.	✓		✓		This is an important feature.
2.	It should correct words(containing repeating letters), that are not present in dictionary	✓		✓		This is an important feature.
3.	It should perform stemming efficiently.	✓		✓		This is an important feature.
4.	It should correct short form of words.	✓		✓		This is an important feature.
5.	It should perform sentiment analysis on cleaned data.	✓		✓		This is one of the important feature for calculating final sentiment.
6.	It should use correct antonyms to reverse the tweets and calculate reverse sentiment properly.	✓		✓		This is one of the important feature for calculating final sentiment.
7.	It should calculate dual prediction score properly.The 'alpha' factor should be selected wisely.	✓		✓		This is one of the important feature for calculating final sentiment.
8.	It should calculate polarity ratio and hype correctly .	✓		✓		This is used for predicting final revenue.
9.	The module should store processed tweets in database.	✓		✓		This is important feature.
10.	It should not take too long to process tweets.		✓	✓		This is an important for efficiency of application.

5.2.3 Regression Module

Table 5.4: Unit Testing

Sr. No.	**Test Cases**	**Critical**		**Test Result**		**Comments**
		Yes	**No**	**Accept**	**Reject**	
1.	Module should calculate final regression accurately .	✓		✓		This is an important feature that will produce final revenue prediction of the movie.
2.	Module should store the revenue calculated in the database for respective movie.	✓		✓		This is an important feature.

5.3 Integration Testing

Table 5.5: Integration Testing

Sr. No.	Test Cases	Critical		Test Result		Comments
		Yes	No	Accept	Reject	
1.	User click on a movie poster should call appropriate function to display prediction.	✓		✓		This is an important feature of final product.
2.	Data processing module should be called after collection of data.	✓		✓		This is one of the important for sentiment analysis of collected data.
3.	Data processing module should call regression module after processing data.	✓		✓		This is important for calculating revenue.
4.	Sentiment Analyser should receive data in correct format.(cleaned data).	✓		✓		This is one of the important feature for calculating final sentiment.
5.	Regression module should get correct values of regression coefficients of the movie.	✓		✓		This is an important feature.
6.	Administrator should be able to call data collection , data processing and insertion of movie details separately	✓		✓		This enables admin to show predictions of upcomming movies.

Chapter-6
Results

The below given are the test results of the Box Offcice Revenue Prediction Application on three test movies. Polarity ratio is calculated that gives the ratio of positive tweets to negative tweets. Hype is calculated as ratio of distinct user to total number of tweets. Rate of tweets is added to it. Similarly, other factors such as actor , actress, genre, sequel and holiday effect are considered. Finally, Revenue is calculated.

Table 6.1: Result

Movie Name	Polarity Ratio	Hype	Actor	Actress	Sequel	Genre	Holiday	Revenue
Jai Gangaajal	7.2	0.5	0.4	1	1	1	0	89 Cr.
Ki and Ka	7	0.5	0.6	1	0	0.8	0	99 Cr.
Kapoor and Sons	10	0.6	0.8	1	0	1	0	134 Cr.

Table 6.2: Comparison

Movie Name	Predicted Revenue(Cr)	Original Revenue(Cr)	Error
Jai Gangaajal	89	62	27
Ki and Ka	99	16 (Going on)	-
Kapoor and Sons	134	124 (Going on)	-

Table 6.1 shows the various factors contributing in prediction of revenue. The movie ”Jai Gangaajal” had polarity ratio 7.2, which is considerably good. Also, it was a sequel and had a high rated genre. But the movie suffered low hype and had no holiday on the day of its release. Considering all these factors, regression is carried out and a total revenue of 89 Cr was predicted by the system. The movies actual Box office collection is 62 Cr.

Movie ”Kapoor and Sons” had a good polarity ratio, star-cast and genre. A total revenue of 134 Cr is predicted by the system. Its actuall collection is 124 Cr and still counting. Movie ”Ki and Ka” also had a considerably good polarity ratio, star-cast, average hype and no holiday effect. Its predicted revenue is 99 Cr and actual collection is 16 Cr and still counting. These two movies are still running in theatres and hence their actual revenue values can change.

Accuracy of the system increases as more training data is available. The system proves to be efficient in predicting the Box office revenues of movies by minimizing error. Table 6.2 shows

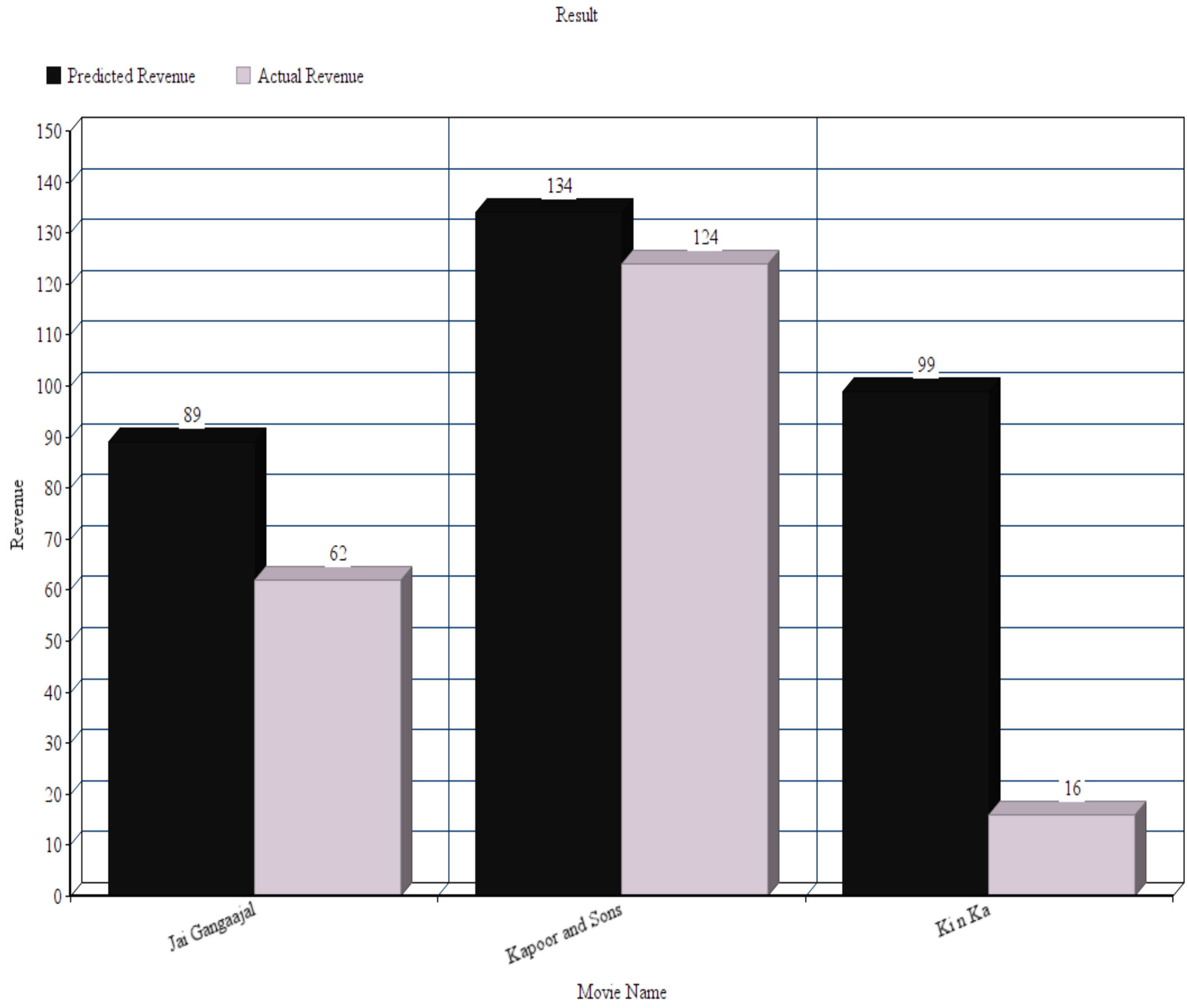


Figure 6.1: Result Chart

the predicted as well as actuall revenue collection of the above mentioned movies. Figure 6.1 represents the same.

Chapter-7

Conclusion

Thus, the proposed system, Box Office Revenue Prediction using Dual Sentiment Analysis proves more efficient. DSA overcomes the drawbacks of traditional systems by addressing the polarity shift problem.

The accuracy of system is increased by taking into account other factors such as sequel, genre, star-cast and holiday effect. The results will become more accurate as more training data is available. This system can be extended to various domains such as election, sports, consumer goods etc, prediction.

References

[1] ui Xia , Nanjing, China ; Feng Xu ; Chengqing Zong ; Qianmu Li “”Dual Sentiment Analysis : Considering Two Sides of One Review”, Knowledge and Data Engineering, IEEE Transactions on (Volume:27 , Issue: 8).

[2] . Prodromidis and S. Stolfo,“Pruning Meta- classifiers in a Distributed Data Mining System, Proc. First Natl Conf. New Informa- tion Technologies, Editions of New Tech. Athens, 1998, pp. 151160.

[3] . Abbasi, S. France, Z. Zhang and H. Chen, ”Selecting attributes for sentiment classification using feature relation networks”, IEEE Trans. Knowl. Data Eng., vol. 23, no. 3, pp.447 -462 2011

[4] . Kim and E. Hovy, ”Determining the sentiment of opinions”, Proc. Int. Conf. Comput. Linguistic, pp.1367 -1373 2004

[5] afeng Lu, Robert Kruger, Dennis Thom, Feng Wang, Steffen Koch, Thomas Ertl, and Ross Maciejewski, ”Integrating Predictive Analytics and Social Media” , DOI: 10.1109/VAST.2014.7042495 Conference: IEEE Conference on Visual Analytics Science and Technology (VAST).

[6] . Thomas Barthelemy ,Devin Guillory ,Chip Mandal , ”Using Twitter Data to Predict Box Ofce Revenues”

[7] I. Councill, R. MaDonald and L. Velikovich , ”What’s great and what’s not: Learning to classify the scope of negation for improved sentiment analysis”, Proc. Workshop

Negation Speculation Natural Lang. Process., pp.51 -59 2010

[8] . Dave, S. Lawrence and D. Pen-nock, ”Mining the peanut gallery: Opinion extraction and semantic classification of product reviews”, Proc. Int. World Wide Web Conf., pp.519 -528 2003

[9] . Hu and B. Liu, ”Mining opinion features in customer reviews”, Proc. AAAI Conf. Artif. Intell., pp.755 -760 2004

[10] . Kim and E. Hovy , ”Determining the sentiment of opinions”, Proc. Int. Conf. Comput. Linguistic, pp.1367 -1373 2004