Group 6

MovieProphecy

Software Requirements Specification

Yogesh Bodhe, Abhishek Chakraborty, Zachary Chapman, Aravinda Yempada
12/7/2014

MovieProphecy Requirements Specification

Version 1.1

December 7, 2014

Revision History

Name	Date	Reason For Changes	Version
Group 6	10/9/2014	Initial document release	1.0
Group 6	12/7/2014	Revisions as of Sprint 3 completion	1.1

Document Signoff

Name	Date	Signature	Role
Bodhe, Yogesh	12/7/2014		Contributor
Chakraborty, Abhishek	12/7/2014		Contributor
Chapman, Zachary	12/7/2014		Contributor
Yempada, Aravinda	12/7/2014		Contributor
Dr. Mohsen Dorodchi			Project Supervisor

Table of Contents

1. Executive Summary

1.1 Project Overview

Inspired by the work of Vasu Jain in his 2013 paper "Prediction of Movie Success using Sentiment Analysis of Tweets," we have proposed MovieProphecy, a new take on the aggregate review website model. Our system will make calls to Twitter's developer API to gather data that we will then run through a hierarchical sentiment analysis with the help of LingPipe. Once analyzed, we will have a polarized review of a movie (positive, negative or neutral). Anyone who is interested in movies can access the front end website of our system in order to search particular movies for a review to help determine if it is something they would like to see.

1.2 Purpose and Scope of this Specification

In this specification we will describe early development features of the MovieProphecy system, including the back end systems as well as the front end web application. The intended audiences of this documentation are business stakeholders and system designers and developers.

In Scope

This document addresses requirements related to the first three sprints of our project:

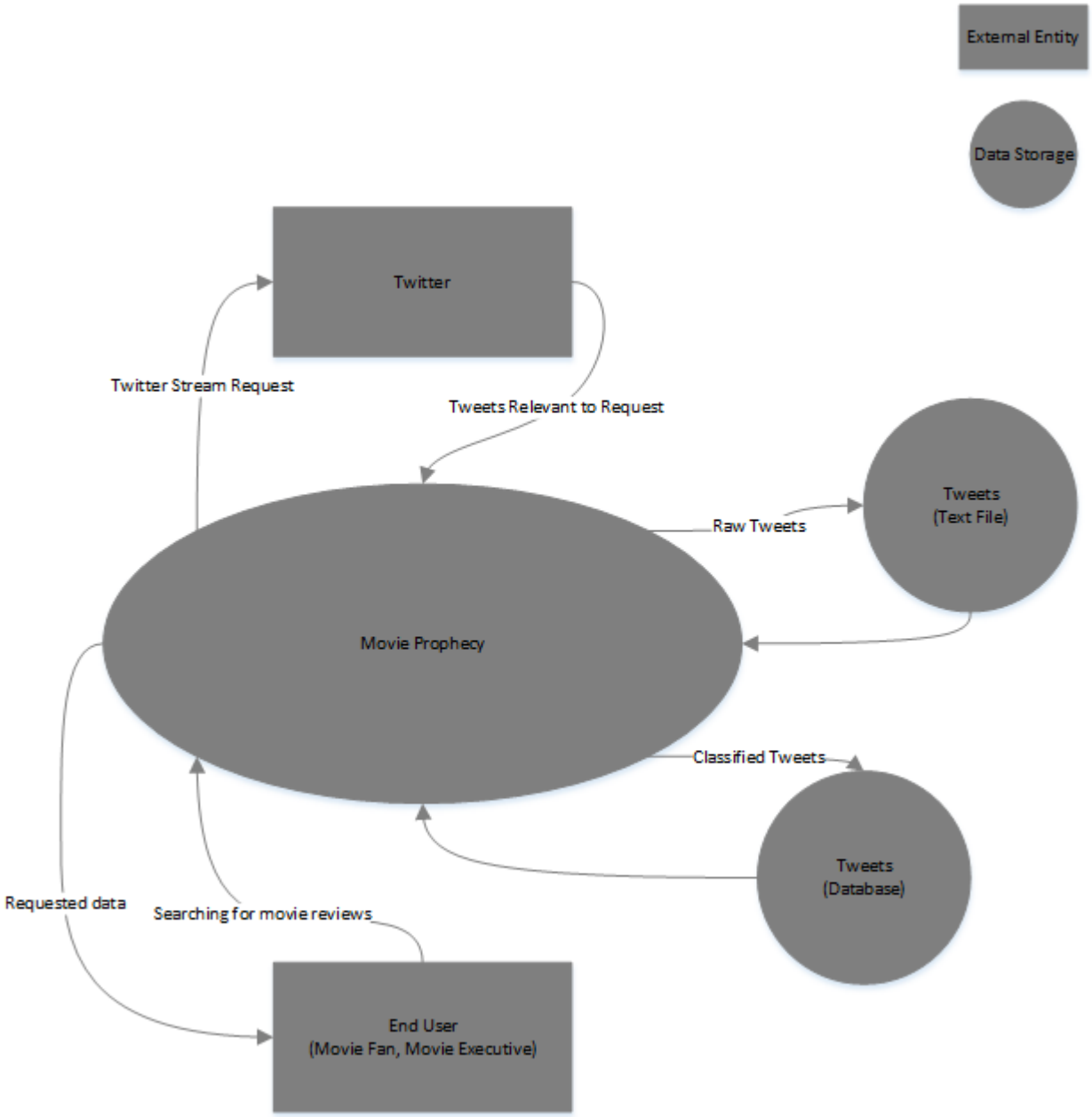
- Creation of our data gathering and sentiment analysis systems
- Creation of user facing front end systems (web browser application)

2. Product Description

2.1 Product Context

User generated content is reshaping the way in which people interact with each other. As more people opt to social media for speaking their minds, 'opinion mining' is of great interest to individuals and organizations. A small 140 character Twitter submission (Tweet) has a wealth of information which, if analyzed, can provide deep insight into people's minds. We are interested in scrutinizing tweets about movies in order to predict their popularity.

Our system will obtain Tweets as data to perform a sentiment analysis to determine if a movie is "good" or "bad." Basic users will be able to access this review in order to make a decision as to whether or not to see the movie while professional users can access additional information.



2.2 User Characteristics

End users can be classified into two different categories dubbed “Movie Fans” and “Movie Executives.”

Movie Fans

- These are individuals who seek online movie reviews to help influence their viewing choices.
- Low level of movie knowledge and experience needed for use of the product, but may have more.
- Only low technical expertise needed to access basic functionality of the product.

Movie Executives

- These are individuals who wish to see more information about movies than just basic reviews such as user opinion over time.
- Higher levels of movie knowledge and experience or a desire to research opinion information.
- Low to moderate technical expertise needed for additional product functionality.
- Access same information as “Fans” but with added functionality.

2.3 Assumptions

- Twitter continues to offer and support its developer API
- Twitter remains a relevant force in social media

2.4 Constraints

- Growing need for disk space as we store Twitter data
- Only current Tweets and those created in the past week can be gathered
- User access and security for those who sign up for “Executive” service
- Operation with Twitter’s API as our primary source of data
- Mixing software tools and languages needed for different system parts

2.5 Dependencies

- Requires frequent data collections calls to Twitter’s streaming API
- LingPipe sentiment analysis is required to be implemented to the project for reviews

3. Requirements

Priority Definitions

The following definitions are intended as a guideline to prioritize requirements.

- Priority 1 – The requirement is a “must have.”
- Priority 2 – The requirement is needed for improved functionality for the majority of users
- Priority 3 – The requirement is either “nice to have” or additional functionality for professional users.

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_01	The system should request new Tweets to act as text in our analysis.	Twitter4J (for use with Java)	1	9/18/14	

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_02	The system should store Tweets in a text file dedicated to the movie it is in reference to.		1	9/18/14	
BR_03	The system should perform a sentiment analysis on each text file to produce a review for each movie	LingPipe hierarchical analysis	1	9/18/14	
BR_04	Fans should be able to search for specific movies in order to get a review		2	9/18/14	
BR_05	Fans should be able to see reviews of movies.	Basic product functionality for end users	1	9/18/14	
BR_06	Fans should see a list of upcoming and recent films		2	9/18/14	
BR_07	Executives should see specific opinions that contribute to the review	Sampling of “good” and “bad” Tweets	3	9/18/14	
BR_08	Executives should be able to see a film’s box office performance		3	9/18/14	
BR_09	Executives should be able to see opinions over time		3	9/18/14	
BR_10	Executives should be able to sign in with credentials to access additional functionality		3	9/18/14	

3.2 User Interface Requirements

During our initial build we will solely be focusing on a desktop web browser interface. Later on in the product life span there can be adaptations made for mobile browsers and applications as needed. At the basic level, users will be able to search the site for movies and be given details, including the sentiment review, based on their selection.

3.3 Usability

The user web interface itself is very simple to use. Movies can either be selected from a list or searched by title. Once a movie is selected, standard users will see a rating based on how many Tweets are positive, negative, and neutral. “Executive” users will be able to see more data such as user opinion over time and a selection of Tweets gathered.

3.4 Performance

3.4.1 Capacity

Potentially hundreds or thousands of users should be able to access the site simultaneously. Most content displayed will be static with the only calculations performed from user input being page redirects and “Executive” sign-ins.

3.4.2 Availability

Our user interface will ideally remain available at all times. Most of our calculations and maintenance will be focused on the back end systems that generate reviews and should remain independent of the website's availability.

3.5 System Interface/Integration

Text files will be used to store raw Tweets for our classifier to access. Once analyzed, Tweets and their classification will be stored in a SQL Server 2012 database. Twitter API makes data requests to Twitter on behalf of the system admin. A web interface interacts with the user to get the name of the movie and display its review.

3.6 Security

3.6.1 Protection

All the movie fans will be able to use the application directly. Movie executives, on the other hand, have to register themselves and log in into the system to get access to the underlying data analysis details of the system, for their business purposes.

3.6.2 Authorization and Authentication

The system uses OAuth (Open Authentication) standard to gather tweets using the Twitter API. It is a three-way handshake protocol, which authenticates the user on twitter and authorizes him/her to access the Twitter data. It is more secure than passwords, as passwords are more vulnerable to theft.

3.7 Data Management

Types of information used by various functions

We store the following attributes for a particular movie:

- tweet_id
- tweet_text
- username
- time_of_tweet
- method_of_tweet
- location

Frequency of use

For sentimentally analyzing tweets our major area of focus would be the attributes named 'tweet_id', 'tweet_text', 'time_of tweet' and 'location'.

Data access rules

Only system developers have access to the data.

Data entities and relationships

Movie tweets and their attributes.

Integrity constraints

Primary key in our database would be 'tweet_id'.

Data retention

We store all possible tweets and their attributes in our database for optimizing our analysis potential.

Data formats

- tweet_id – long integer
- tweet_text - varchar

- username - varchar
- time_of_tweet - timestamp
- method_of_tweet - varchar
- location - varchar

Default or initial values

None. All values are ground truth data.

3.8 Portability

The system can be ported to different OS environments like Windows, Linux, and Mac OS X as it uses open source technologies like Twitter API and LingPipe sentiment analyzer. The web user interface of the system can be accessed from any digital device. A specialized user interface for mobiles and tablet devices will be implemented in later builds.

4. User Scenarios/Use Cases

Use Cases and User Stories can be viewed in the following Google Drive folder:

Use Cases/User Stories
(https://drive.google.com/folderview?id=0B0emyt296qGiVzBrMlBOQUlFbms&usp=sharing)

5. Deleted or Deferred Requirements

Identify any requirements that have been deleted after approval or that may be delayed until future versions of the system. For example:

Req#	Business Requirement	Status	Comments	Pri	Date Rvwd	SME Reviewed /Approved
BR_08	Executives should be able to see a film's box office performance	Deferred	Time constraints		12/1/14	

APPENDIX

Appendix A. Definitions, Acronyms, and Abbreviations

Define all terms, acronyms, and abbreviations used in this document.

API	Application Programming Interface – a set of routines, protocols and tools for building software applications.
IMDB	Internet Movie Database – a website that keeps records of movies, possible candidate for us to obtain lists of upcoming movies.
Java	Programming language used for the majority of our back end systems.
LingPipe	Java library for classifying text in numerous ways, used in our project for sentiment analysis.
SQL	Structured Query Language – special-purpose programming language for managing data held in a relational database management system.
Twitter	Social media site where users may publish 140 character posts for the world to see.
Twitter4J	Twitter API library for use with the Java programming language.
Tweet	A single post on Twitter.
OAuth	Open Authentication – three-way handshake protocol used by the Twitter API to authenticate a user on Twitter and authorize them to access data.
OS	Operating System – software layer between the hardware and user applications.

Appendix B. References

http://www.jscse.com/papers/vol3.no3/vol3.no3.46.pdf

Use Cases/User Stories
(https://drive.google.com/folderview?id=0B0emyt296qGiVzBrMlBOQUlFbms&usp=sharing)