

Project report



September 28, 2014

Sai kishore bandaru

EBENEZER ANAND ARAPALLY

**USAura: Sentiment Analysis Application Using Storm**

**Summary:**

Each person has his own state of mood like Happy, Sad, Angry, Excited etc. Now to broader the horizons it is important to know about the country’s mood too at one point of time. So our application gives the scope of analyzing the current mood of the country based on twitter tweets and videos watched on the YouTube in a specific hour of a day.

**Design:**

In order to analyze the trend of the data that has been retrieved we used our own algorithm to classify the tweets. Our classification can recognize thirty different adjectives that define the mood of an individual. The data collection is planned to collect from both YouTube and Twitter.

**Data Model:**

The data that is being collected from twitter is stored in a text file with comma separated fields. The fields of the data consists of tweet, location and unique twitter ID. However we couldn’t be able to collect the data from YouTube as the APIs we invoked weren’t retrieving the data. We tried to retrieved text data from YouTube which consists of highly viewed videos in a respective location.

**Integration Model and Algorithm:**

The Initial design of our application consists of three steps. They are:

1. Initially we will be collecting the twitter data from storm. Inherently we also store this data in a file on the local machine.
2. Now the data which we have collected will be uploaded to HBase for storage as it can hold schema less column oriented data storage.
3. Now from the Android application we will be retrieving data from the HBase and analyzing the data to detect the mood in the country.

The algorithms used for the application are basically the classification algorithm, which has been developed on the basis of basic recursive algorithm. The algorithm initially searches for a key word say “happy”. Now the tweet will be further searched for words like “not happy” i.e. some meaning which meant opposite to the keyword that has been searched initially. In this way we shall be classifying the tweets into sad, happy, angry, excited etc.

The following diagram depicts the brief overview of the integration data model of our application:

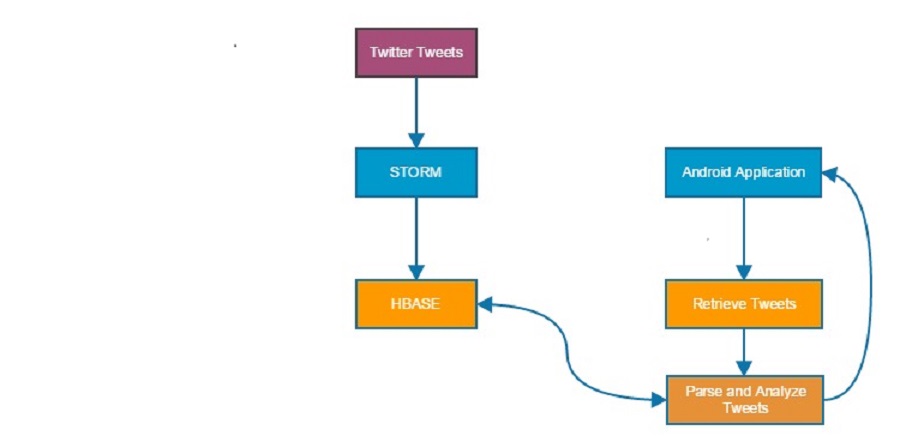


Figure 1Design and Architecture

Initially we tried using the Naïve Bayes algorithm using Weka i.e. Weka java APIs. But since we couldn’t execute the results properly. We were supposed to prepare our own set of algorithm to test the data. We have prepared certain trained data and give it for Weka Java APIs as part Naïve Bayes classification. We were able to train data but unable to test data properly. These data has been uploaded in the Google Cloud for storage .

**Selection of Datasets:**

The datasets are being collected solely from twitter and are stored in HBase. As long the storm topology continues we will have the data to be inserted in the HBase. Keeping in view about the memory limits of server machine we will write only 1% tweets into a file at a time and will upload them to storage repository i.e. HBase. The following table shows the database model of HBase:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Row | Column Family(Tweets Analysis) | | | Timestamp |
| RowID | TweetID | Location | Tweet |  |
|  |  |  |  |  |

**Software Specifications:**

Tools: Storm, Android SDK, Google Web Services (Visualization)

Development OS: Widnows8, Android, Linux

Mobile Deployment: Android

Databases: HBase

**Implementation:**

**Classification algorithms:**

The data that has been inserted in HBase will be retrieved initially using the RESTful services. Now we shall classify these data based on the tweet content. This is purely a context based classification but not a collaborative based classification. We mount all the results and display the results. In storm topology, we have made implementation in such a way that when a spout collects data from twitter we are simultaneously writing the data into the file. Whenever the spout proceeds to a bolt for further execution at this point of time we are making the HBase calls in order to store the data. As a result of continues processing of both spout and bolts we have data being collected and also being pushed in real time.

The retrieval drawback we had in our application is we are making the pruning of data inside the mobile application i.e. the timestamp of the tweet is being checked in the front end which might be a costly execution when we consider huge amounts of data.

**Front End Application :**

The application is a Native Mobile application and has been developed for Android users. The application supports from Android 4.2 version to Android 4.4.2 version. And the SDK support ranges from 11 to 18. We also used the REST web service technology in order to insert and retrieve data from HBase.

**Visualization:**

For projecting the analyzed information we have used Google Web Services. We used the Google Play services as part of data visualization. We have represented the data in the form of Column Charts and Pie Charts in the mobile application.

**Outputs:**

Initial mobile page consists of a button, which on click will give you the tweets made from an hour. The following snapshot depicts the main page of the application:

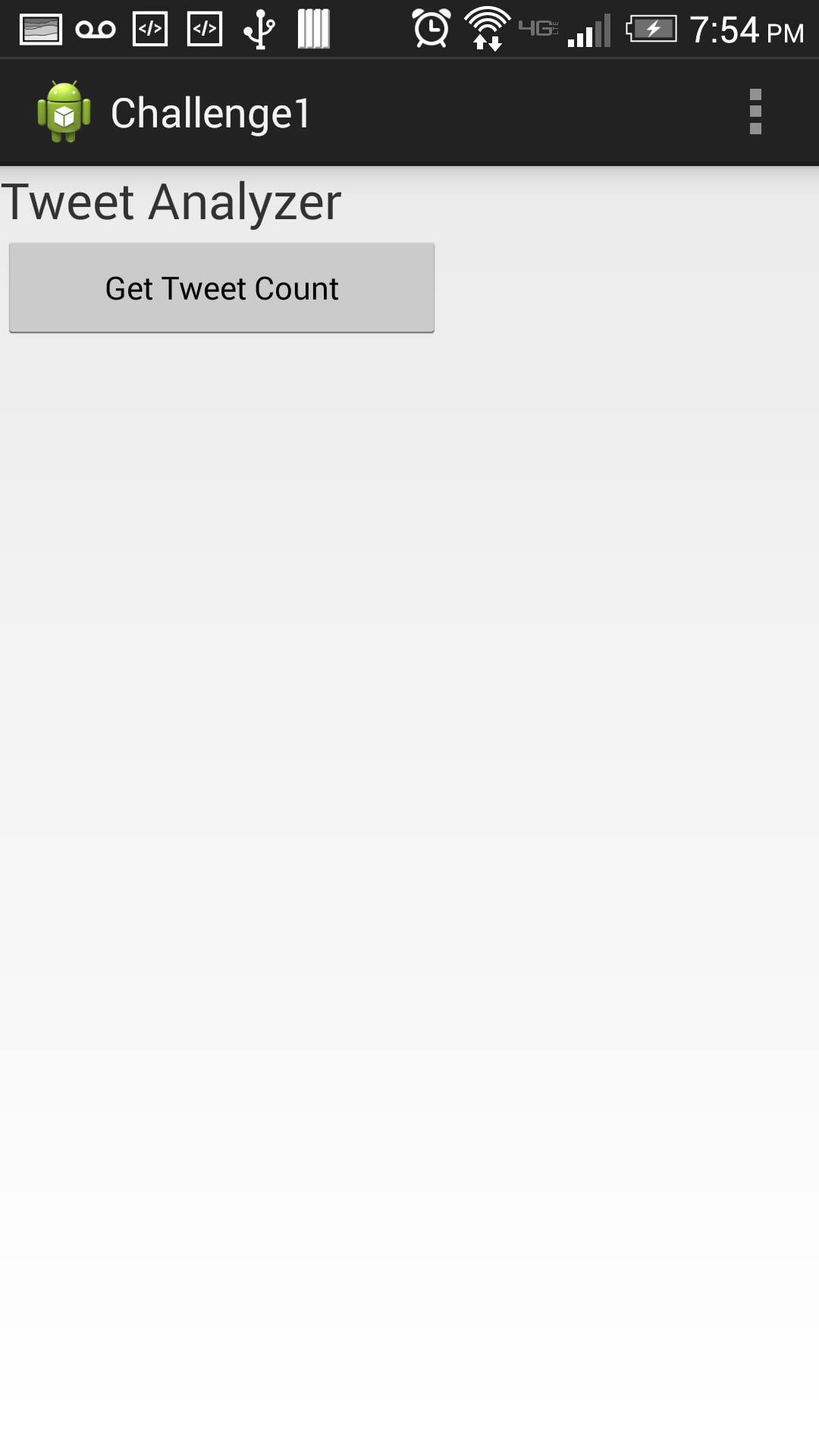


Figure 2Main Web Page

Now when you click on GetTweetCount you will get the result and also you have a button “Analyze Tweets”. The following snapshot gives you the visualization of the page:

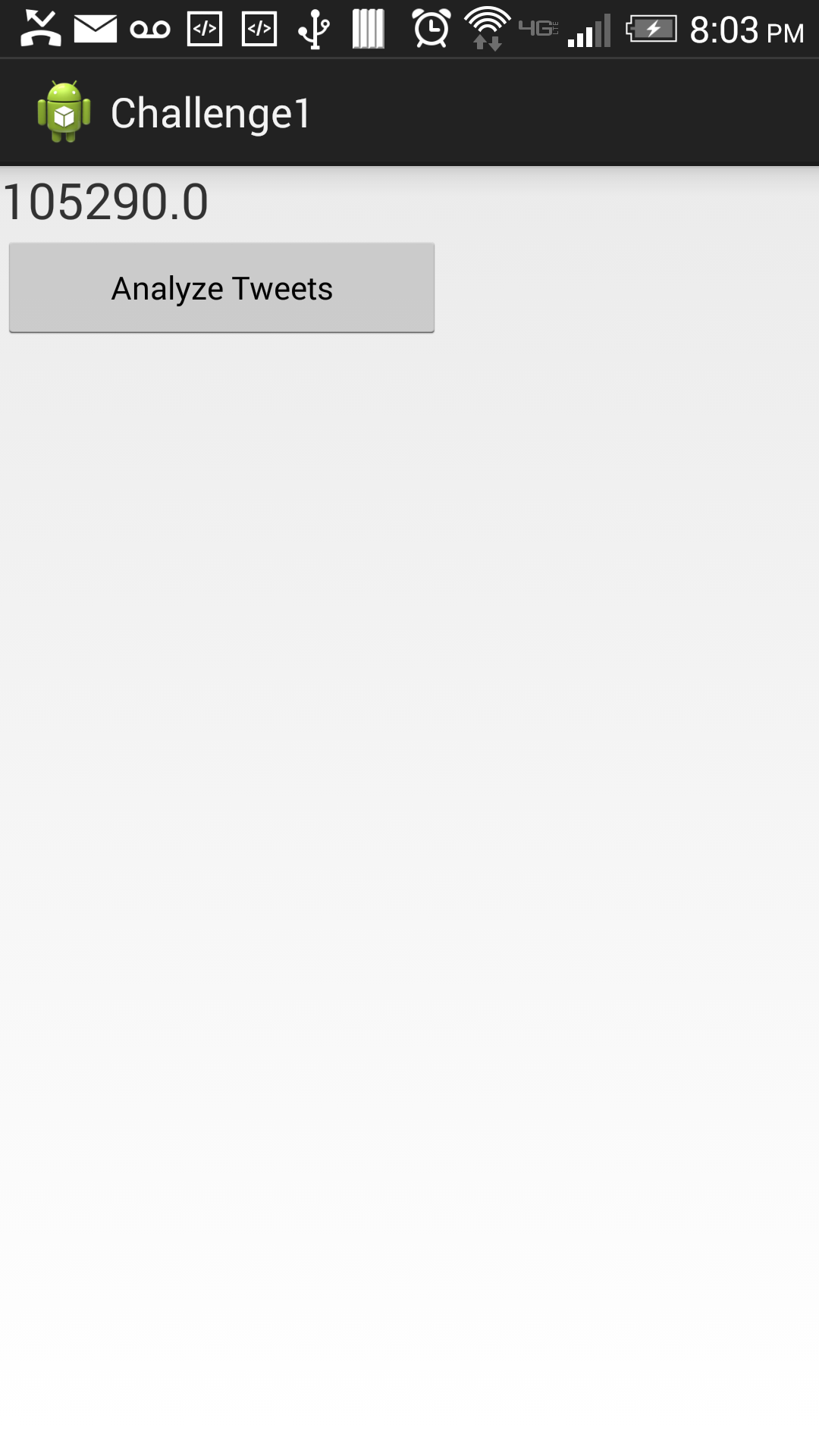


Figure 3Tweets count

On clicking the Analyze Tweets button you will be displayed with analysis of percentage of people in different moods in a pie chart and column chart representation.

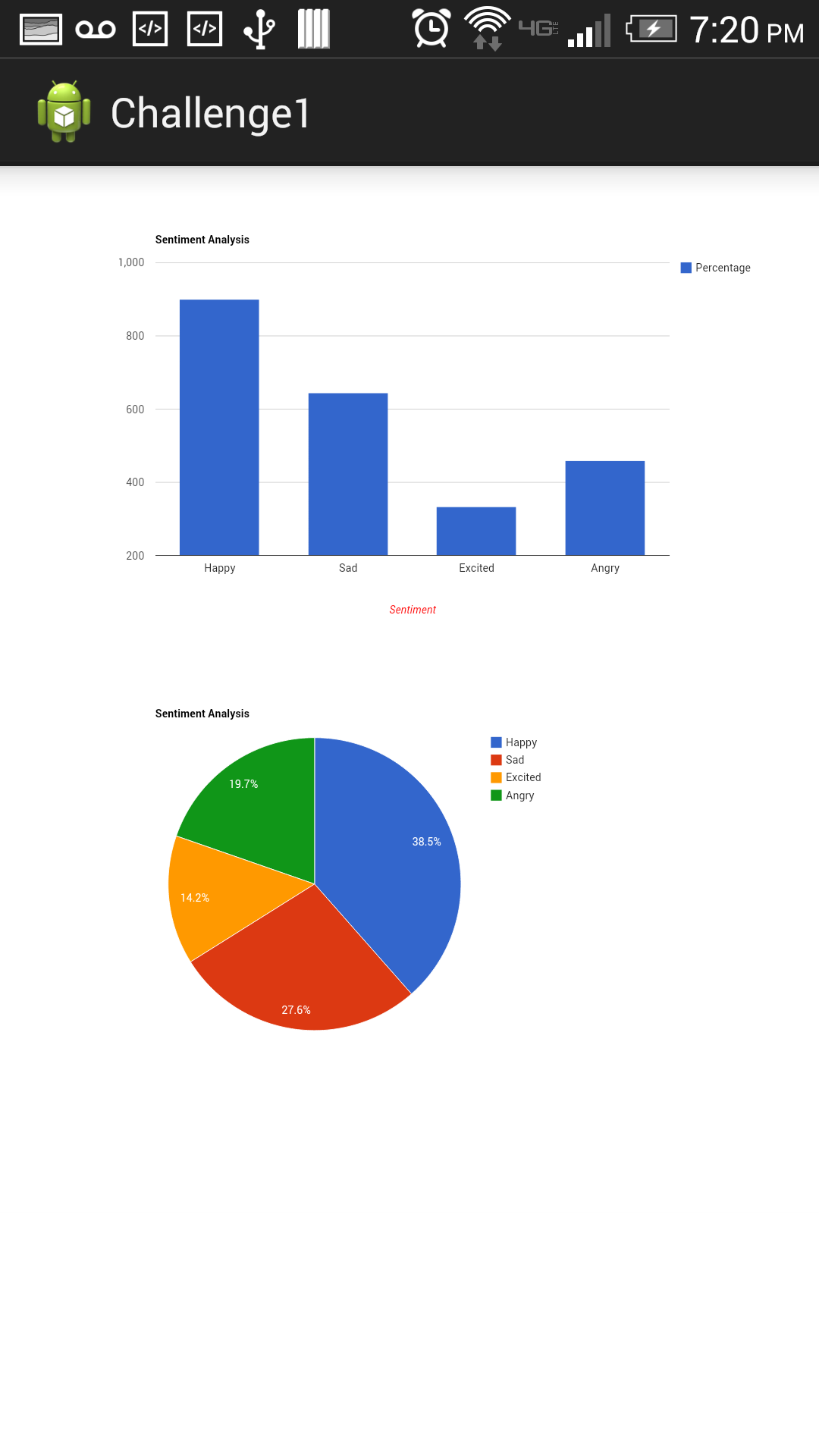


Figure 4Tweet Analysis

The following is the snapshot of our storm topology:

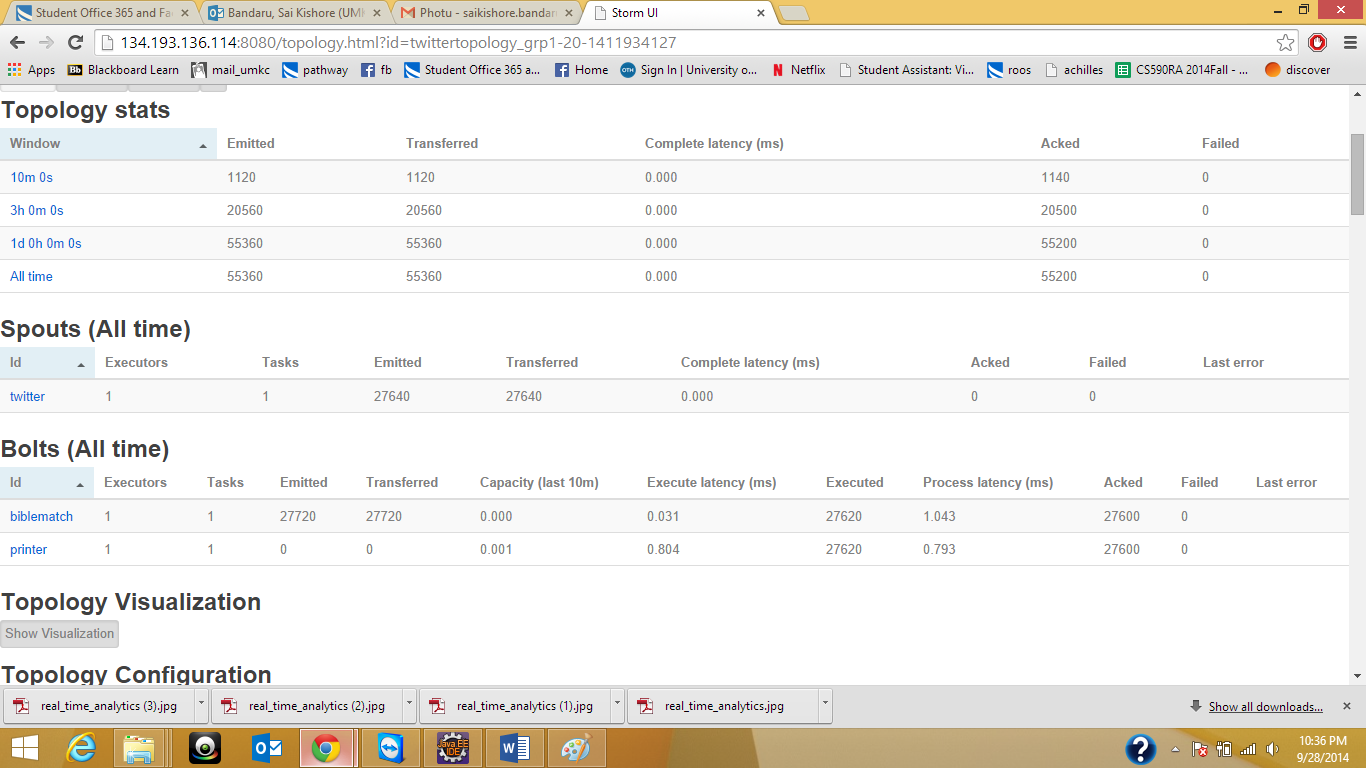


Figure 5Storm Topology

**GitHub URL:**

The following is the github link for entire code implementation:

* <https://github.com/SaiKishoreBandaru/Challenge1_RA>

**Limitations:**

The limitations of this project are we couldn’t be able to write the tweets consistently using storm i.e. through bolt implementation. And the Naïve Bayes classification through Java Weka was not also feasible so our application can distinguish only 30 different words in a tweet to classify into a particular emotion. And also we were not able to retrieve data using the YouTube APIS streams provided by Google

**Enhancements:**

* Data analysis through YouTube
* Recommendations based on the localities to revive based on collaborative filtering.

**References:**

* HBase integration in Storm - <http://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.5/bk_user-guide/content/ch_storm-using-hbase-connector.html>
* YouTube streaming - <https://developers.google.com/youtube/v3/getting-started>
* Analysis of Tweets - <http://schoolofdata.org/harvesting-and-analyzing-tweets/>