Design Document
Sentiments Analysis with Twitter
Team Jazz Men

Anagh Goswami 1217426
Meet Pandya 1214306
Jasman Gill 1211554
Jesse Truong 1222722
Jia Xu 1213268

April 9, 2017

Contents

List of Figures

List of Tables

1 Revision History

Table 1: Revision History

Description of Changes	Author	Date
Created first version with appropriate sections.	Meet Pandya	January 7, 2017
Edited the document	Jesse Truong	January 8, 2017
Worked on Solution to Challenges	Anagh Goswami	April 7, 2017
Edited for Final Revision	Meet Pandya	April 8, 2017

2 Introduction

2.1 Purpose

The purpose of Sentiments Analysis with Twitter is to provide a way for high school students to see what is being said about a university on the social media. This project also provides benchmark analysis which would help student compare and contrast two potential universities of their choices.

2.2 Description

Twitter is a hugely popular social media application used to communicate people's thoughts and opions with those around them. There are currently a whole generation of students that use the application to see what others think of a certain topic. Some of these students being highschool seniors who are trying to decide what university they are going to attend. There is no medium for them to be able to quickly assess their fellow peers thoughts on certains schools other then doing independent research. Allowing students to see collected and analyzed data would greatly help them in being able to determine which university is right for them. This is even more so true for students who are on the edge of making a decision, but would like to see others opinions be it negative or positive to help them with their own. Depending upon how succesful the project is, it could possibily even be used by companies to determine how they can improve and what others around them are doing that is seen as positive features.

2.3 Scope

The scope of Sentiments Analysis is started with but not limited to High school students. This has a potential to be expanded to organizations, brands and many more categories where Sentiments Analysis could be utilized for the intended purpose.

3 Development Details

3.1 Language of Implementation

Sentiments Analysis is based on the Client-Server architecture style as the inputs to the system comes from users which are the clients and the resulting scores are retrieved from the server using the APIs. The source code of the Sentiments Analysis is written with Python using the Flask Web Framework. The Webpage UI used to display the results is developed using HTML, CSS, Bootstrap Framework and Javascript.

3.2 Supporting Frameworks/APIs/Library

Alchemy Language API
Twitter API
Requests Library for HTTP (Python)
Flask Web Framework
Bootstrap Framework

4 Implementation Components

4.1 Flask Function

Flask Web Framework is used to build the webpage using python. The way Flask works is that you define functionality at each page of the Web UI that you are working on. The Web UI is hosted on localhost and pages are defined in Flask from there based on the route. Each page for the Web UI on Flask has a function that it executes when it is on the page.

The "home" function within the code for Flask which is used for the homepage calls upon the HTML file that encapsulates all the Front End code for the homepage of the UI. The HTML file is the start point of the application. And similarly the other functions like cutom page and login page are called upon when the user has navigated to those respective pages.

Eventually the website will be hosted live on a web server of the team's choice. During the developmental stage, the web app is locally hosted. In Figure 1, the psuedo-code displays the Flask Functions.

```
Define Homepage:
    render (home.html)

Define Custom Page:
    render (Custom.html)

Define Login Page:
    Check if user is not already logged in
        Verify user information is valid
        Authenticate password
    render (login.html)
```

Figure 1: Flask Function

4.2 Get Data Function

The "getdata" function as defined, iterates through the query "keyword" and each letter is fed as the input to the Twitter Search Function described in the succeeding section. Each letter is inputted in the Twitter Search function and the resulted data is added to a "storage variable". By the end of the loop, the "storage variable" has all the data retrieved from the Twitter Search Function. Finally, the "storage variable" is what is returned in this Get Data Function. The functionality of Get Data can be seen in Figure 2.

```
result = []
Define Get Data (topic):
    if length of result is 0:
        Loop through the topic:
            Call Twitter Search Function with each letter of the topic
            Add the results from twitter search to result
            Loop again until going through all the letters of the topic

    return result
```

Figure 2: Psuedo-Code of Get Data

4.3   Twitter Search Function

In this section the twitter search function will be discussed that entails using the Twitter API and also the Alchemy API. The twitter search function has an input parameter which is the query 'keyword' to be searched for on twitter. This keyword is used as the query in the URL for the Twitter API. Requests library is used to get the tweets from twitter API and the resulted data is stored in a variable using JSON format as seen in Figure 3.

```
Define Twitter Search (topic):
    url = 'https://api.twitter.com/1.1/search/tweets.json?q=topic'
    Call Authenticate to access the API

    Request to get 100 tweets in English
    Save the results in Data
```

Figure 3: Twitter Search Function

4.3.a   Alchemy API

Within the Twitter Search Function, the Alchemy API is called upon to feed the tweets in order to retrieve the Sentiment score. The score is tallied based on its emotion whether it is neutral, positive or negative. Neutral scoring tweets are not accounted for in the final result as they do not make a difference. The snippet of code for the Alchemy API is provided in Figure 4.

```
Loop through Data:
    Filter the tweets by removing emojis for compatibility with Alchemy API and ASCII
    filter_text = item[text].encode('ascii','ignore')

    Analysis = ['id': item, 'tweet': text, 'user':username, 'followers':followers count]

    Sentiment = alchemylanguage.sentiment(text)
    If error occurs:
        print exception
    If sentiments[status] OK:
        Analysis[sentiments] = sentiments(type)
        If Analysis[sentiments] = Neutral
            Neutral Count = Neutral Count + 1
        Else
            Analysis[Score] = Sentiments[Score]
            TweetList[Analysis[Sentiments]].Add(analysis)
            TweetList[Score] += Sentiments[Score]


TweetList[Positive].Sort
TweetList[Negative].Sort

Return TweetList
```

Figure 4: Alchemy API function

4.4 Store and Retrieve Tweets

The functions for Storing and retrieving tweets from the database are described in this section. These functions are part of the Data Object module. The Storing function deals with saving tweets in the database with the topic that it pertains and to and also the date of the entry. While the retrieval function deals with simply getting the tweets from the database and putting it into an Array. The figure below shows the psuedo-code for the Store and Retrieval of tweets.

```
Define Save Tweets:
    Loop through TweetList
        subject = TweetList[subject]
        date = TweetList[date]
        db_data = Collection.find(subject, date)


        If db_data = 0:
            Collection.insert(TweetList)
        Else:
            pass

Define Get Tweets(subjects, collection):
    Result = []
    Loop through Subjects
        res = Collection.find(subject, date)
        If res = 0
            res = Nothing
        Else
            res = res[0]
            res[id] = Nothing
        Result.add(res)
    Return Result
```

Figure 5: Store and Retrieve Tweets

4.5   Personalized Data

The functions for Personalized Data also come from the Data Object module. Custom
data is saved when a user searches for a topic using the search bar, once they have
logged in. The resulting data is then stored to the database and it is associated with
that user in particular. Also, for the retrieval of the data, the function looks for
any data associated with the currently logged in user. If there exists any data, that
will be displayed or else no data will be shown. Figure below shows the high level
implementation of the two functions.

```
Define SaveCustomData (data, username)
    data['username'] = username
    db['customdata'].insert(data)

Define GetCustomData (username)
    result = []
    res = db_data['customdata'].find['username':username].sort['date']

    if res = 0:
        res = Nothing
    else
        topics = Nothing
        Loop through res:
            if res['subject'] Not in Topics:
                topic.add(res['subject'])
                result.add(res)
```

Figure 6: Personalized Data

4.6   User

The functions for User define the User Registration and User Retrieval attributes of the system. User Registration checks whether a user already exists with the same username or not, if then only does it register the new user. User Retrieval simply checks through the database if a user exists and then adds it to a list of array.

```
Define Create User(user)
    user = db_data[user].find('username':user)
    if user = 0:
        db_data['users'].add(user)
        print "User Created"
    else:
        return "user already exists"


Define Get User(username)
    res = db_data[user].find('username':user)
    If res = 0:
        res = Nothing
    else:
        res = res[0]

    return res
```

Figure 7: User

5 Error Handling

5.1 Server Failure

- Backup and Recovery Scripts

- Condition to redirect traffic when server is down

6 User Interface Elements

In this section the different elements of the Web UI are displayed with images and brief description.

6.1 Home Page

This is the top of the Home Page where the user begins from every time using the application. After clicking on the "Tell Me More" button, the webpage traverses down to the next section. The side menu allows for the traversing of the home page. By clicking on each button, the page navigates automatically to that particular section of the homepage.
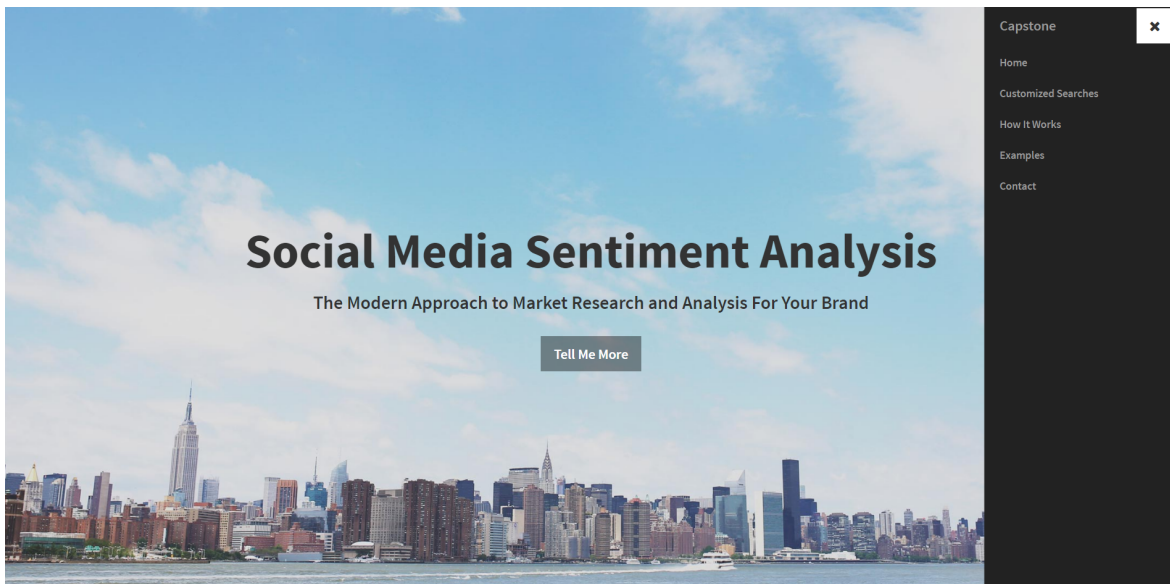

Figure 8: Top of Home Page

6.2 How It Works

After scrolling down on the home page, this section known as "How It Works" gives a brief description of how the Sentiments Analysis works.
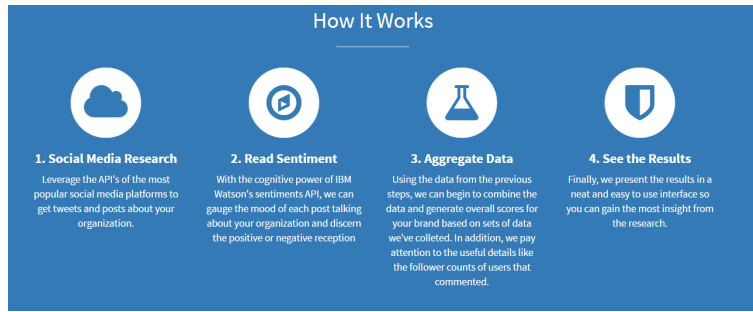
Figure 9: How it works

6.3   Demo

After further scrolling down on the home page, the Sample section shows four images with Categories on top of each. By Clicking on the University Category, it displays 4 pre-defined objects within the category. The three images below show the process step by step. First image shows the 4 pre-defined categories. The next image shows the 4 objects within the University Category. The final image shows a table with Most Positive and Most Negative tweets.
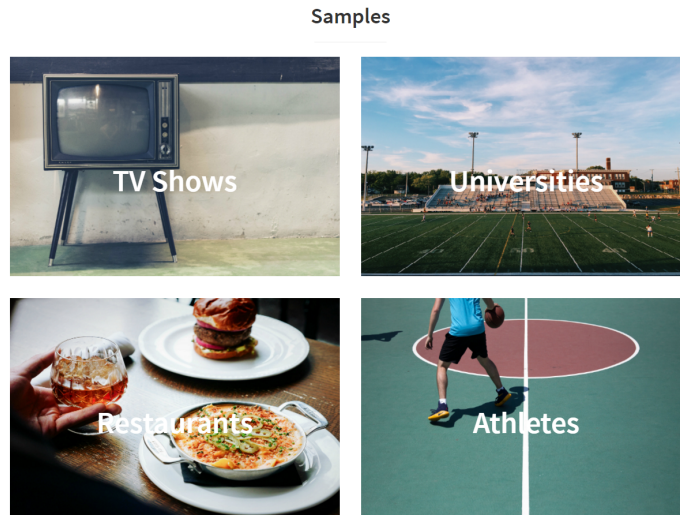
Figure 10: Demo

Figure 11: Demo 2
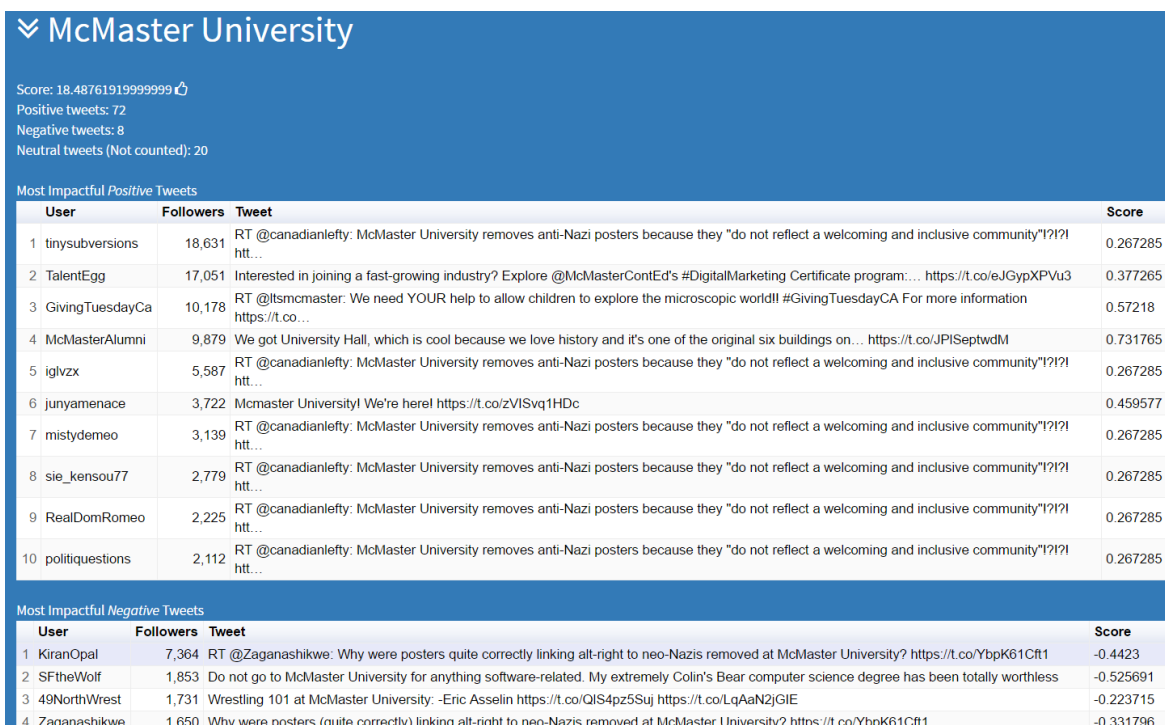
Figure 12: Demo 3
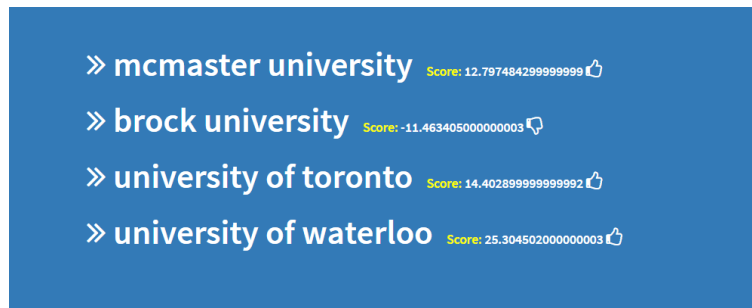
6.4 Custom Search

The Custom Search page can be accessed by using the Right side Menu, and there is a button that says "Customized Searches". After arriving on the custom search page, we can see in the figures below, that at first you arrive at a login page. And once logged in after registering with a user name, there is a search bar allowing the user to look for any topic of their choice.
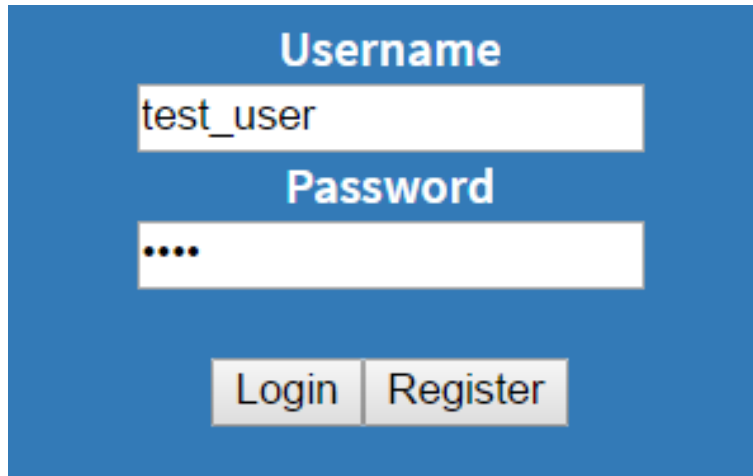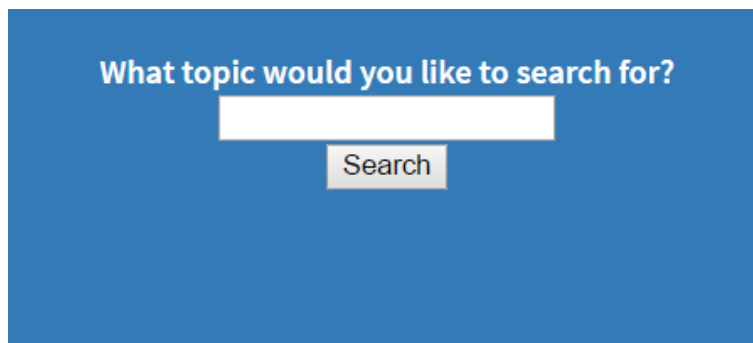
Figure 13: Custom Search

Figure 14: Custom Search 2

6.5   Custom Data View

The Custome search result data can be displayed in two manners. One is in a bubble and one is in table. The bubble view shows the most impactful positive and negative tweet. While the table shows 10 Most Positive and 10 Most Negative tweet about the topic searched by the user. Figures below show the two different views.

Results

Refresh data

≫ **samsung** Score: 4.73117 👍

Display Mode:
Bubbles | Tables

Positive tweets: 9
Negative tweets: 1
Neutral tweets (Not counted): 24

Most Impactful Tweets

RT @bahmboozle: Making the switch from @Tmobile to @verizon this is going to be interesting. Also buh-bye Apple Hello Samsung S8 Plus #

ilovemygear
Followers: 2579   Score: 0.848359

Finally gave in and pre-ordered the Samsung #GalaxyS8 phone. It's no Note 4 replacement, but the battery life on this tiny S6 is atrocious

garth_hamilton
Followers: 4465   Score: -0.874619

Figure 15: Bubble View

Positive tweets: 14
Negative tweets: 15
Neutral tweets (Not counted): 63

Most Impactful *Positive* Tweets

	User	Followers	Tweet	Score
1	LavieTheCockato	2,252	RT @SamsungSLife: Live smarter with #SamsungConnect. Control your smart devices and appliances, right from your Galaxy phone. https://t.co/	0.811954
2	brackster39	1,791	Finally got mine yesterday. Have mixed feelings about buying another Samsung, cause of the locked bootloaders . https://t.co/0CGaGIzMHB	0.78935
3	ECresthillUK	632	#BOGOF #Sale Mobile Smart Phone Stand For All Brands iPhone Samsung Great Gadget https://t.co/EHRQRVQAC8 @eBay	0.855517
4	BoopTheCat	572	https://t.co/oVYpd6fej is giving an awesome #Samsung #GalaxyS8 mobile. Participate NOW! @shubham_noo #Giveaway #S8 https://t.co/h9SxXkK52a	0.613726
5	GamerMonkey_	475	RT @SamsungSLife: #SamsungDeX turns any space into a workplace. Connect your phone to the DeX Station & get to work. #SamsungSmartLife http	0.283015
6	DavidFoyn	279	RT @MoMo_SanDiego: Well I have a Samsung and it's still locking up and even quitting on me. https://t.co/7GoIm15Qfl	0.475886
7	galliumoutlyin	254	Samsung Galaxy S6 Screen Protector Made with Ultra Clear PET Plastic Gives You Protection From. https://t.co/7FfTmqzoPL	0.568154
8	freydasales	231	RT @ECresthillUK: #BOGOF #Sale Mobile Smart Phone Stand For All Brands iPhone Samsung Great Gadget https://t.co/EHRQRVQAC8 @eBay	0.849178
9	rc4f	187	Best Bet (Upgrade)Eazymate Fashion Running Belt Stylish Travel Money Belt with Zipper and Smartphone Pocket https://t.co/1o6zqBB4Lm	0.60411
10	sales_trudy	77	RT @ECresthillUK: #BOGOF #Sale Mobile Smart Phone Stand For All Brands iPhone Samsung Great Gadget https://t.co/EHRQRVQAC8 @eBay	0.849178

Most Impactful *Negative* Tweets

	User	Followers	Tweet	Score
1	jspring86az	13,875	Shout out to Samsung for finally reimbursing my Exynos Note 7 return...I missed my $900 these last 5 months! https://t.co/Gqenlr1NsY	-0.796468
2	BugattiMusicOn	13,395	Samsung Attacks DMN for Suggesting They Manufactured Those Exploding Headphones https://t.co/30XyWQ6yqY https://t.co/Z1s7UtZie	-0.546283
3	3rdrockhome	7,946	This might be the first look at Samsung's follow-up to the exploding Galaxy https://t.co/FGC5hykNvz #Rumor #Leaks https://t.co/CPatKC40FO	-0.864239
4	LawGro	3,721	Samsung Chief trial: lawyer flatly denies charges https://t.co/RMl6aSLBeWH #lawyer	-0.497429
5	Hawke525	854	RT @RerezTV: You may see this image floating around of a burnt Nintendo Switch from a blown up battery. This is fake it's actually a Samsun	-0.675836
6	Bigrig480	787	Finally retired my #Lumia phone. Bought a Samsung S7. Too much stuff being dropped by Microsoft... Ruined a brilliant phone.	-0.680524
7	Glass_Eaters	381	RT @RerezTV: You may see this image floating around of a burnt Nintendo Switch from a blown up battery. This is fake it's actually a Samsun	-0.675836
8	Trintus	341	RT @RerezTV: You may see this image floating around of a burnt Nintendo Switch from a blown up battery. This is fake it's actually a Samsun	-0.675836
9	Scopher_	212	RT @RerezTV: You may see this image floating around of a burnt Nintendo Switch from a blown up battery. This is fake it's actually a Samsun	-0.675836
10	gopiz_world	122	View through Samsung Note 4. View within. Taken by Google Pixel. No filter was required! https://t.co/N4QlxoSFD	-0.370155

Figure 16: Table View

7   JSON Data Format

The data set of searched tweets are collected in JSON format. Currently, the data set is updated according to the interest of the user. Below is an excerpt showing the JSON format from the data collected through the Twitter API:

```
[
  {
    "negative": [
      {
        "followers": 11678,
        "score": "-0.579868",
        "sentiment": "negative",
        "tweet": "RT @yung_mung: @bendykoval pretty sickening how forgotten Suharto's legacy has become  by the Western left.
Always on about Pinochet but ne\u2026",
        "user": "bendykoval"
      },
      {
        "followers": 11624,
        "score": "-0.619831",
        "sentiment": "negative",
        "tweet": "RT @Aristokles11235: Pardon me for standing up for western civ against the impending idocracy, far left OR
faux right.\nOn second thought do\u2026",
        "user": "Costofles"
      },
      {
        "followers": 9458,
        "score": "-0.743297",
        "sentiment": "negative",
        "tweet": "RT @chrisderrick1: The Tainted Dollar \nTellin' a story the way the Old West used to be\n
https://t.co/3NIfJmvnZV\n#JakeBase #Texas #western\nR\u2026",
        "user": "JSYorkshire"
      },
      {
        "followers": 3291,
        "score": "-0.760624",
        "sentiment": "negative",
        "tweet": "RT @SoniaKatiMota: Republicans call UniversalHealthcare Communism except they fail 2 add all Western
Civilizations have, except f*ckin' Ame\u2026",
        "user": "QlaraQontra"
      },
      {
        "followers": 2063,
        "score": "-0.366874",
        "sentiment": "negative",
        "tweet": "RT @singofhisgrace: @LiteraryLucifer @RT_com Yes, it does. Our western news isn't accurate when compared
worldwide. Not by MILES.",
        "user": "LiteraryLucifer"
      },
      {
        "followers": 1924,
        "score": "-0.328925",
        "sentiment": "negative",
        "tweet": "You can't be a Nationalist in most current-day Western countries and a Catholic. It's simply not possible.",
        "user": "GreyWarden"
      },
      {
        "followers": 1923,
        "score": "-0.429771",
        "sentiment": "negative",
        "tweet": "RT @waffen2112: Environmentalist should be called what they really are, subversionist!  Another affliction
western countries have to deal w\u2026",
        "user": "waffen2112"
      },
```

Figure 17: JSON data format

8   Algorithms

This project does not have many complex algorithms. This is because the Alchemy API from IBM solves our complexity. // The way the Alchemy API is utilized is by feeding a tweet as an input to the API. The tweet is then checked to be assigned a Positive or Negative type by the Alchemy API. Once a positive or negative type is assigned to a tweet, the Alchemy API then assigns the Sentiments Score to the particular tweet that is being executed in the on-going iteration. If the tweet gets a Neutral tag then there is a counter variable keeping track of neutral tweets which is increased by 1.

The resulting sentiments data is stored in a JSON data object by splitting the tweets based on their tags, whether they are positive or negative. The neutral tweets are not stored anywhere, but rather only the total number of neutral tweets is stored separately.

9   Solution to Challenges

9.1   Challenges

Following is the challenge addressed in the Problem Statement:

The challenges for this project would be figuring out the way to go through social media and getting all the necessary information needed based on a topic or a keyword. We have to work with the information that is freely available on the internet. One of the major issue with social media is all the information can be biased, so the information we collect has to be presented in a manner that it does not come across as factual. Also, picking what kind of organizations to focus on is a challenge because not all organizations are actively discussed on social media platforms.

9.2   Solution

The first challenge addressed was finding the means through which to go through social media data. This was solved by utilizing the Twitter API, which provides many different ways of retrieving tweeted data. Data is returned in JSON format so not very user friendly way of displaying. Therefore, the data will be presented in a systematic and understandable manner.

The second challenge addressed was the bias involved with the information from social media. Because of this reason, the amount of data to be collected in order to do the analysis will be in big sizes in order to have lesser impact from biases.

The third challenge addressed is what kind of organizations to focus on. This was solved by leaving the choices of organizations or institutions to be openly decided by the end user.

10 Traceability Matrix

A traceability matrix is provided in the table below to show links between requirements and modules.

Table 2: **Traceability Matrix**

Requirement	Module(s)
1	5
2	5
3	3,8
4	2,6
5	2,6
6	2,6
7	1,7
8	7
9	7
10	3
11	5
12	1
13	3
14	4
15	5
16	9
18	9
20	9