Analyzing Emotions - Twitter / Reddit / Youtube

Nagalakshmi Prasanna Pujita Bodapati
Computer Science
Binghamton University
Binghamton, New York, USA
nbodapa1@binghamton.edu

Saisuraj Aitha
Computer Science
Binghamton University
Binghamton, New York, USA
saitha2@binghamton.edu

kshamitha Gandu
Computer Science
Binghamton University
Binghamton, New York, USA
kgandu1@binghamton.edu

Mohith kumar Sopparam
Computer Science
Binghamton University
Binghamton, New York, USA
msoppar1@binghamton.edu

Anshul Upadhyay
Computer Science
Binghamton University
Binghamton, New York, USA
aupadhy5@binghamton.edu

Abstract

The comments sections of social media platform have become the new playground for online bullying.The impact of toxic comments is much more catastrophic than we think. Therefore, having a solid toxicity flagging system in place is important if we want to maintain a civilized environment on social media platforms to effectively facilitate conversations.But, deciding if we have to flag a comment or not is very time taking. If we have an automated process where we can automatically detect abusive keywords in the comments or posts can save the time of website moderators and also it will have a great impact on improving the discussion rather than focusing on using harsh words. So, we are mainly focusing on posts/comments in Reddit, Youtube and Twitter that contains toxic keywords and flag them as negative emotions by taking into consideration of data we have obtained in a period of time.

Keywords: YouTube API, Tweet-Stream, data analysis, Reddit API, GoogleAPIClient, Python, MongoDB, Twitter API, Reddit API

ACM Reference Format:
Nagalakshmi Prasanna Pujita Bodapati, Saisuraj Aitha, kshamitha Gandu, Mohith kumar Sopparam, and Anshul Upadhyay. 2022. Analyzing Emotions - Twitter / Reddit / Youtube. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Recently, people have started using online forums frequently for discussions. Along with the discussions on online forums, trolls and spammers have become more common. It is a time-consuming and tedious job to moderate the comments and posts on these forums, so the organizations have to rely on external people to handle them. We are trying to differentiate between negative and positive comments and tweets on Twitter, Reddit, and YouTube. To achieve this, we would get real-world data from the above mentioned platforms. We also intend to visualize Twitter, Reddit, and YouTube's data.

2 DATA COLLECTION

For analyzing the twitter data ***Tweetstream*** module is used. This module will provide a framework where Twitter streaming API can be used. For scraping Reddit data - we used ***RedditAPI*** that is posted on Reddit can be accessed using this module. For the third data source we have choosen YOUTUBE and the data can be accessed using a google ***GoogleClientAPI***

2.1 COLLECTING TWITTER STREAMING API

Tweetstream is a python module that can be used to acquire tweets from the Twitter streaming API. Mainly there are two kinds of API's to obtain the tweets from the twitter - Samplestream and FilterStream. . In real time FilterStream is mainly used as it sends the tweets by considering a mere set of criteria and this criteria can filter the tweets by the following 3 keys:

- To search for certain specific keywords or can also track by a specific keywords.
- Based on the current user it can filter user specific tweets.
- This also can filter using specific locations but the tweets should be geo-tagged tweets.

We developed the twitter data collection using java. By accessing the various secret API keys and bearer tokens collected the data using the URI Builder ***tweets/sample/stream***. Once the data is collected we used python modules -***textblob*** and ***numpy*** by filtering the posts using regex. The following are the posts that have been collected. The overall data collection is **235197**.

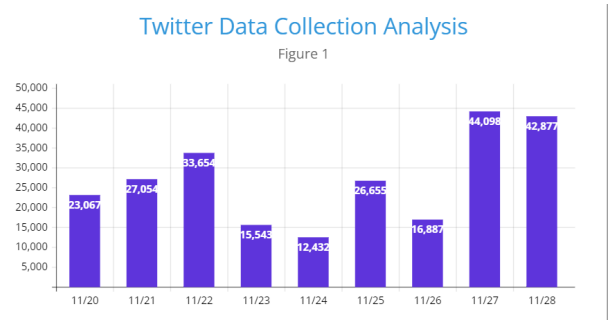

Figure 1. shows the overall collection of twitter for specific dates

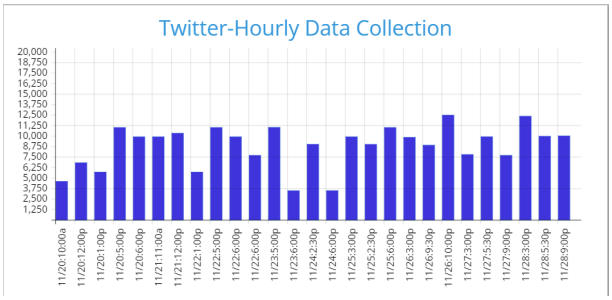

Figure 2. shows the hourly collection of twitter for specific dates

2.2 COLLECTING YOUTUBE STREAMING API

For extracting YOUTUBE comments we use GOOGLECLIENTAPI that is mainly designed for Python client-applications. By using this API it is very flexible and also easy to access Google API's. There two types of access - Simple or Authorized. To use one of there access methods we have to make use of a build() function that will generate a service object and then from here we can decide if we need Simple access or Authorized access. By taking random videos we will collect all the random comments and later we analyze them.

- Read only instance can be used to retrieve the post that are publicly available.
- Where as Authorized instance can be used to post, comment etc.

Once the data is collected and stored for a frequency of dates in mongodb collections the following is the final analysis of comments collected for a period of time. The overall data collection is **66603**.

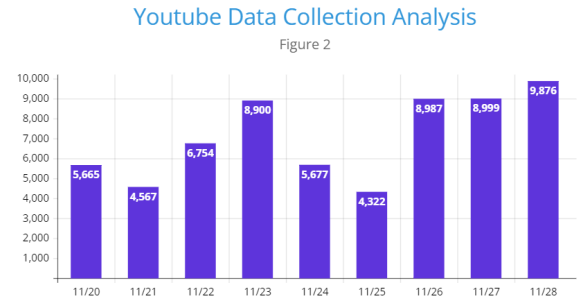

Figure 3. shows the overall collection of youtube comments for specific dates from a video with more than 60Million comments

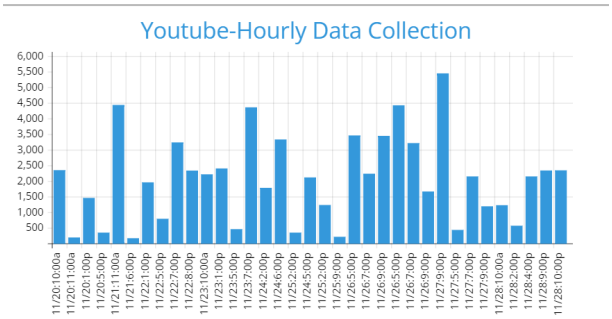

Figure 4. shows the hourly collection of youtube comments for specific dates from a video with more than 60Million comments

2.3 COLLECTING REDDIT STREAMING API

For extracting posts that are posted on Reddit we are using Reddit API where we directly access the data by using HTTP requests uisng ***www.reddit.com/dev/api/***. We are mainly using Python programming language to make the HTTP requests and store the extracted data in Mongodb using the module - pymongo. Also, we have created triggers where the data will be automatically collected and will store in the mongodb.

Once the data is collected and stored for a frequency of dates in mongodb collections the following is the final analysis of posts collected for a period of time. The overall data collection is **262312**.

3 DATA COLLECTION ESTIMATES

For twitter we are collecting approximately 1M tweets, for Youtube - approximately 6M comments and for Reddit we

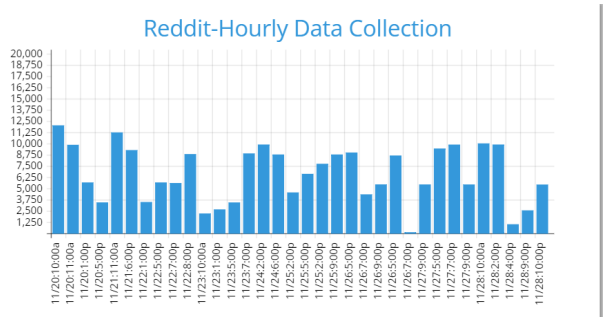

Figure 5. shows the overall collection of reddit posts for specific dates from the subreddit r/politics

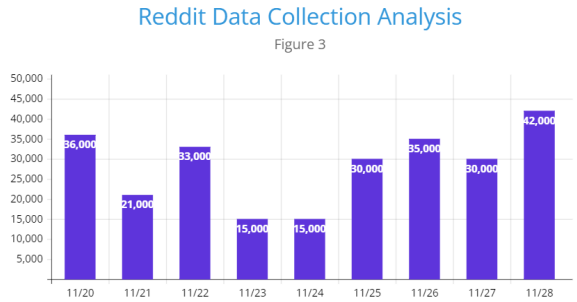

Figure 6. shows the hourly collection of reddit posts for specific dates from the subreddit r/politics

have an option of passing the parameter "LIMIT" and can set any value we wish to collect, so right now we are thinking of collecting approximately 100 posts per day. The data that is acquired is meaningful and we can use this data to analyze the emotions.

4 METHODOLOGY

We used SampleStream to gather tweets from twitter, GoogleAPI-Client to extract random comments for a particular video and used direct Reddit oauth authentication and directly hit the API to collect data on subreddits by generating a query and collected the posts from these subreddits. Once, the complete data is collected we stored in the MongoDb by creating separate database and collection for each datasource we have chosen.

We have collected an average of 235197 twitter posts and approximately 66603 YouTube comments and around 262312 reddit posts over the past seven days. The data will be then extracted from MongoDB and cleaned by implementing a regex function. By using this function, we will now remove the noise from the data collected. For example, this function can be used to remove data that is not in english language. Now we use this updated clean data to perform the analysis to find emotions whether a particular tweet/comment/post is positive, Negative or neutral.

The process of final analysis or result can be obtained using Textblob which is python module. By check every comment/post that is collected and sent to textblob the result will be in form of a polarity where we have 3 options - -1, 0, 1 which shows Negative / neutral / Positive.

5 RESEARCH OBJECTIVES

The main aim of the project is to understand better how the people use the social media sites. The decisions made by these individuals may come from the situations or individual persons attitude and mindset. Its not unusual to come accross hateful comments or posts that are made by large number of persons.

The main research question we are trying to find is:

- From the collected data - what percent of data is POSITIVE? - From the data collection part and the performance of sentimental analysis for all the 3 data sources, we found that for Twitter, the positive percentage is **13.69**, and for Youtube, the positive percentage is **93.01**, and for Reddit, the positive percentage is **12.27**
- From the collected data - what percent of data is NEGATIVE? - From the data collection part and the performance of sentimental analysis for all the 3 data sources, we found that for Twitter, the negative percentage is **5.25**, and for Youtube, the negative percentage is **6.98** and for Reddit, the negative percentage is **5.36**
- From the collected data - what percent of data is NEUTRAL? - From the data collection part and the performance of sentimental analysis for all the 3 data sources, we found that for Twitter, the neutral percentage is **5.25**, and for Youtube, the neutral percentage is **0**, and for Reddit, the neutral percentage is **8.64**

For the reddit dataset - the unwanted or filtered comments percent is **73.73**

6 DASHBOARD AND TOOLS ANALYSIS

We are thinking of implementing the dashboard using Pure HTML and for the styling part, we are using Bootstrap. By obtaining the results by performing the sentimental analysis we will graphically represent our result in a web page.

The visualization of the graphical representation will have date pickers where you can select the dates and once the submit button is clicked the corresponding graphs will be shown in the UI.

There are mainly ***two*** data sources analysis - Overall data collected and sentimental analysis for all the 3 data sources.

The following figure is the overview of the inputs taken in the dashboard:

7 TECH-STACK IMPLEMENTED

* 'Python' - For Reddit Data extraction we used python to create and develop this project.

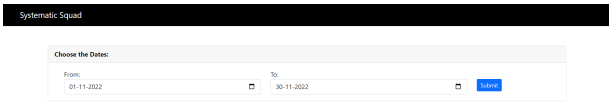

Figure 7. shows overview of the dashboard

* 'pymongo' - For storing comments that are collected from Reddit/Subreddits we used pymongo to conenct to Mongodb client.
* 'flask' - Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.
* 'textblob' - TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.
* 'numpy' - NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.
* 'HTML' - Pure HTML used for rendering browser specific UI.
* 'Bootstrap' - Used for user friendly and interactive styling.

8 HOW TO RUN THE PROJECT?

Install Python
Install Mongodb / Mongoshell
Run server for Mongodb and run mongoshell
Create databases
* 'Twitter' - 'twitterdatabase'
* 'Youtube' - 'youtubedatabase'
* 'Reddit' - 'redditdatabase'
Create Collections
* 'Twitter' - 'twitter'
* 'Youtube' - 'youtube'
* 'Reddit' - 'redditcomments'

'For project 3 implementation run the following command'
ssh -L 5000:127.0.0.1:5000 nbodapa1@128.226.28.120
Password to enter: 'MEeaaNe92'

Then to run the UI
paste the URL in the browser - http://127.0.0.1:5000

9 ANALYSIS FEATURES

Based on the inputs selected, we implemented two analysis where we show the overview of all the data that is collected in form of a bar graph and for each data source we have performed sentimental analysis using textblob module.

The following is the figure generated for the overall data collection:

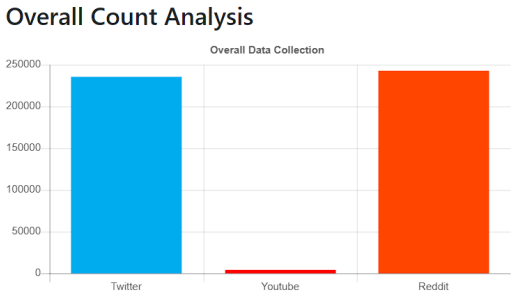

Figure 8. shows the overall data collected by each data source.

The following is the figure generated for the sentimental analysis for each data sources:

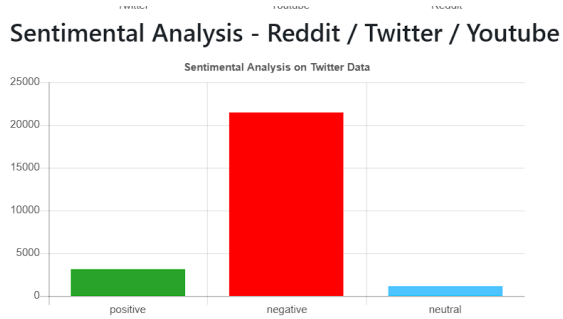

Figure 9. shows the sentimental analysis on Twitter

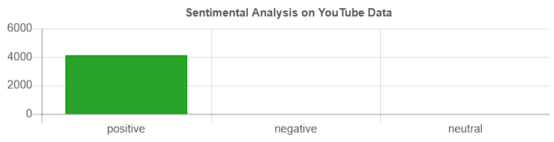

Figure 10. shows the sentimental analysis on Youtube

10 TERMINAL OUTPUT

The following figure shows the sentimental analysis count obtained for each data source.

11 LIMITATIONS

The application that is developed is not that accurate for all the edge cases. Since, the collection of the data is very huge, performing sentimental analysis on every collection will take a lot of time and can decrease the performance of the application.

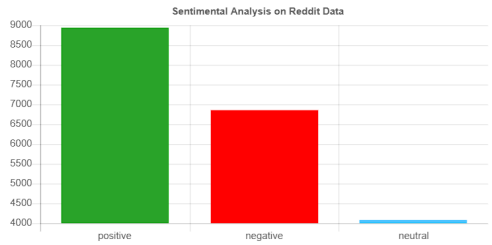

Figure 11. shows the sentimental analysis on Reddit

```
Waiting for analyzing twitter posts
-------------------------------------------------
Twitter positive posts count: 3135
Twitter neutral posts count: 21449
Twitter negative posts count: 1094
-------------------------------------------------
Waiting for analyzing youtube posts
-------------------------------------------------
Youtube positive posts count: 4160
Youtube neutral posts count: 0
Youtube negative posts count: 0
-------------------------------------------------
Waiting for analyzing reddit posts from subreddit r/politics
-----------------------------------------
Reddit positive posts count: 8934
Reddit neutral posts count: 6863
Reddit negative posts count: 4079
-----------------------------------------
127.0.0.1 - - [16/Dec/2022 18:10:37] "POST /result HTTP/1.1" 200 -
```

Figure 12. shows the sentimental analysis on terminal for each data source

12 ACKNOWLEDGEMENTS

We all would like to thank our Professor. Jeremy Blackburn for all the guidance and assistance provided to us for this project. We really appreciate the help and support given to us for the improvement and various opportunities to learn and always explore new technical stuff.

13 REFERENCES

1. Flask app using python - https://www.digitalocean.com/community/tutorials/how-to-create-your-first-web-application-using-flask-and-python-3
2. Textblob - https://www.analyticsvidhya.com/blog/2021/10/making-natural-language-processing-easy-with-textblob/: :text=TextBlob
3. PYMongo - https://pymongo.readthedocs.io/en/stable/
4. Mongodb - https://www.mongodb.com/docs/
5. Bootstrap - https://getbootstrap.com/docs/5.0/getting-started/introduction/
6. Charts - https://chartscss.org/