Sentiment Analysis on Twitter and YouTube

Hemanth Reddy Karri
hkarri1@binghamton.edu
Binghamton University
Binghamton, New York, USA

Punit Paresh Jagani
pjagani1@binghamton.edu
Binghamton University
Binghamton, New York, USA

Akhil Parimi
aparimi1@binghamton.edu
Binghamton University
Binghamton, New York, USA

Vijay Kumar Kadamanchi
vkadama1@binghamton.edu
Binghamton University
Binghamton, New York, USA

ABSTRACT

Toxic comments and personal attacks have become increasingly common on social media platforms, online news commenting areas, and many other public venues on the Internet. However, deciding whether or not to "flag" a comment or post is complex and time-consuming. Not only would automating the process of detecting abuse in comments save website moderators time, but it would also promote user safety and improve online discussions. We're particularly interested in tweets and YouTube comments that contain any harsh or toxic language. We want to identify tweets and YouTube comments that contain toxic phrases and label them as negative using the data we've acquired in a certain time frame.

KEYWORDS

Machine learning models for classification, text mining, text analysis, data analysis, data visualization, MongoDB, Twitter API, YouTube API

ACM Reference Format:
Hemanth Reddy Karri, Punit Paresh Jagani, Akhil Parimi, and Vijay Kumar Kadamanchi. 2018. Sentiment Analysis on Twitter and YouTube. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

The issue of trolls and spammers is getting more prominent as discussions move more and more to internet forums. Manually moderating comments and discussion forums is time-consuming, and organizations are compelled to use contractual or outside moderators to deal with the high volume of comments. We're looking for hate speech on Twitter and YouTube. We consider a tweet or a comment to be hate speech if it incorporates racist or sexist comments. The main objective is to classify negative tweets and comments from other tweets and comments.We'll be collecting real world data from popular media channels like Twitter and YouTube for this project. We'll use the data from these two sources with a defined time frame to discover and highlight any negative sensing content. We also aim to visualize the classified data of Twitter and YouTube.

2 DATA ACQUISITION

The raw tweet data is obtained for Twitter using the python module "tweetstream," which provides a framework for simple Twitter streaming API. The data for YouTube is acquired using the python library "googleapiclient".

2.1 Twitter's Streaming API

We use TweetStream python module that can be used to get tweets from Twitter's streaming API. There are two ways to obtain tweets with Tweetstream: SampleStream and FilterStream. SampleStream merely provides a short, random sample of all tweets that are being posted in real time. FilterStream sends tweets that meet a set of criteria. It has the ability to filter tweets based on three criteria:

- Specific keyword(s) to track/search for in the tweets.
- Specific Twitter user(s) according to their user-id's.
- Tweets originating from specific location(s) (only for geo-tagged tweets).

As we don't have any such constraints for our purpose, we'll use SampleStream mode.

2.2 YouTube API

The Google API Client Library for Python is designed for Python client-applications. It provides easy and flexible access to a variety of Google APIs. All API calls must use one of two types of access: simple or authorized. We utilize the build() function to generate a service object whether we're utilizing simple or authorized API access. Every collection defined by the API is represented by a function in this object. We collect all of the comments of random videos using this method and analyze them later.

3 MOTIVATION

As more and more debates go to online forums, the issue of trolls and spammers is becoming increasingly prevalent. Because manually moderating comments and discussion forums is time-consuming, companies are forced to rely on contracted or outside moderators to handle the huge amount of remarks.

On Twitter and YouTube, we're looking for hate speech. If a tweet or comment contains racist or sexist remarks, we consider it hate speech. The basic goal is to distinguish between bad and

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

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

positive tweets and comments. For this project, we'll be gathering real-world data from prominent social media platforms like Twitter and YouTube. We'll look for and emphasize any unfavorable sensing material using the data from these two sources over a set period of time. We also want to show Twitter and YouTube's categorized data.

4 DATA EXPLORATION

We gather roughly 1 million tweets every day, which equates to 41,667 tweets per hour. We acquired a total of 7162195 tweets (about 7.1 million tweets) and 1839114 YouTube comments based on our data collection rate (approx 1.8 million comments). We have the ability to use every single comment and tweet without screening them since we are not aiming to confine our data to a certain topic. We utilized this information to create useful data that we can use in our study.

5 PROJECT FLOW

Using Python modules TweetStream and GoogleAPIClient, we begin by retrieving real-world data from our two main data sources: Twitter and YouTube. We clean the data once it has been stored in MongoDB to remove all of the noise. Once we have noise-free data, we use machine learning algorithms to develop a classifier (either contextual or general). We now apply the techniques of tokenization, lowercase conversion, and stop-words removal to extract valuable features from the data contained in MongoDB.
Tokenization is the act of breaking down a continuous stream of text into words, symbols, and other significant parts known as "tokens."
Lowercase Conversion: Normalizing a tweet by converting it to lowercase makes it easy to compare it to an English dictionary.
Stop-words removal: Stop words are a group of extremely common terms that, when utilized in a text, provide no new information and are hence considered useless. Examples include "a", "an", "the", "he", "she", "by", "on", etc.
We can then use this data to train the agent once the data matches our expectations. We give the agent points for each successful recognition, and we deduct points if the recognition fails. We transmit our raw data to the agent after it has been trained and tested, and the agent will assess if it is sentimentally negative. After that, we'll evaluate and visualize our findings in a bar graph.

6 BACKGROUND WORK

As more individuals begin to use social media, the amount of data created today is unrivaled. We chose Twitter and YouTube as our data sources since they are the two most prominent social media sites. Since Twitter is open source for data researchers, we were able to create a developer account and use the twitter streaming API "TweetStream" to acquire the requisite tweets, allowing us to capture 1% of all real-time tweets.

We used the "GoogleAPIClient" to collect all of the comments for a random video on YouTube. We've selected a series of videos and gathered all of the comments that have been posted on them.

We use all of the data we collect for sentiment analysis in our

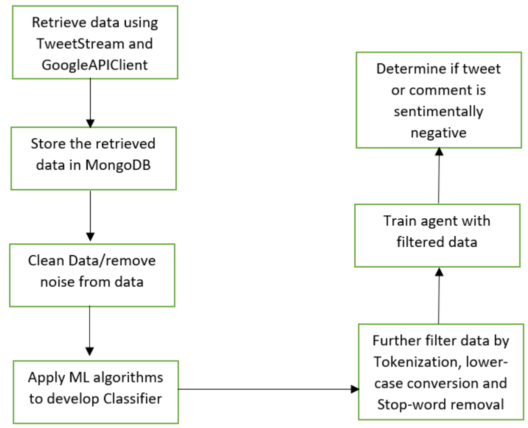

Figure 1: Project Flow

project. We are not screening any of the information we have gathered. Because we are not confining our effort to a specific issue, every single tweet and comment is relevant to us(ref fig.2).

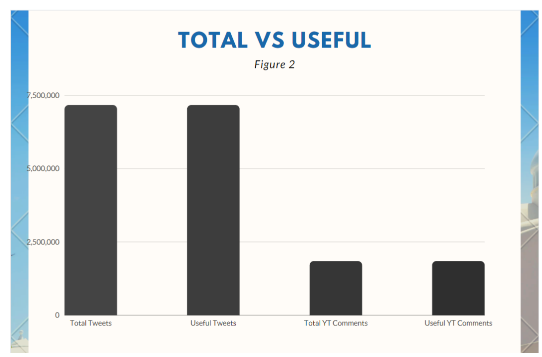

Figure 2: All the tweets and comments that we have are useful for our analysis.

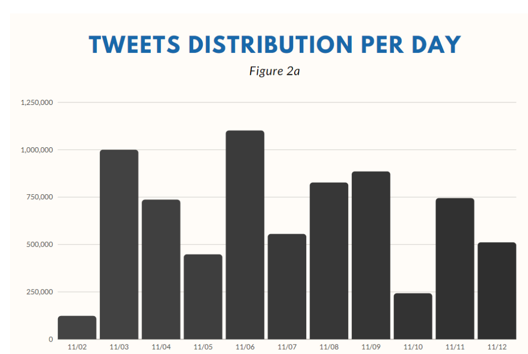

Figure 2a illustrates the number tweets collected per day. X-axis represents the dates and Y-axis represents the number of Tweets.

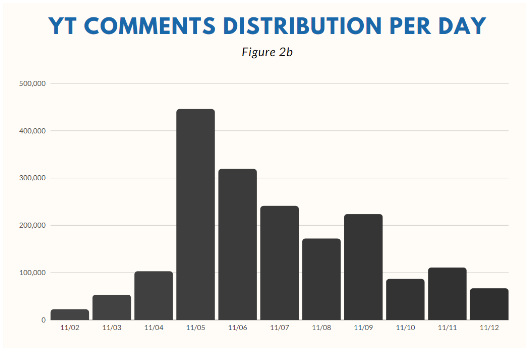

Figure 2b illustrates the number of YouTube comments collected per day. X-axis represents the dates and Y-axis represents the number of YT Comments.

7 RESEARCH OBJECTIVES

The primary goal of this study is to gain a better understanding of how individuals use various social media sites. One may argue that there are many sorts of individuals, attitudes, and situations that influence the decisions made when utilizing the platforms. It's not unusual to come across a large number of hateful comments posted by a large number of individuals.
The three questions we're attempting to answer are:
1) How much of the data is positive?
2) How much of it is negative or offensive?
3) How much of the data is neutral?

8 METHODOLOGY

We utilized the TweetStream python package to gather random tweets from Twitter and GoogleAPIClient to extract comments from YouTube, as indicated in the Data Acquisition section. We saved all of the collected data in MongoDB, from which we will get the data for analysis. We received an average of 7 million tweets and over 1.8 million YouTube comments over the course of seven days. We have the ability to use every single comment and tweet without screening them since we are not aiming to confine our data to a certain topic.

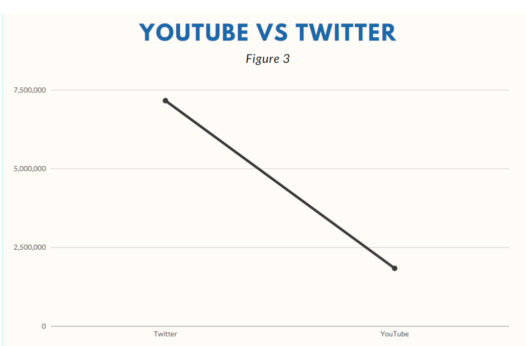

Figure 3 shows comparison between no. of Tweets and YT Comments.

The data is then extracted from MongoDB and cleaned using a regex function. This is how we remove all of the noise from the data. This function, for example, can be used to eliminate tweet content written in a language other than English.

9 API METHODS

For obtaining Tweet data, the Twitter API employs two HTTP methods: GET and POST. In the instance of a Twitter stream, the GET technique is used to connect to the stream. We begin to get a steady stream of information. A succession of delimited JSON-encoded activities, system messages, and blank lines make up the body of the response.
Once we get the required information, we store it in MongoDB.

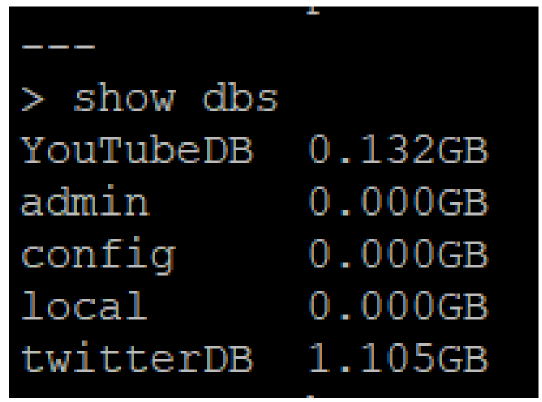

Figure 4 shows the size of DB after the data has been collected.

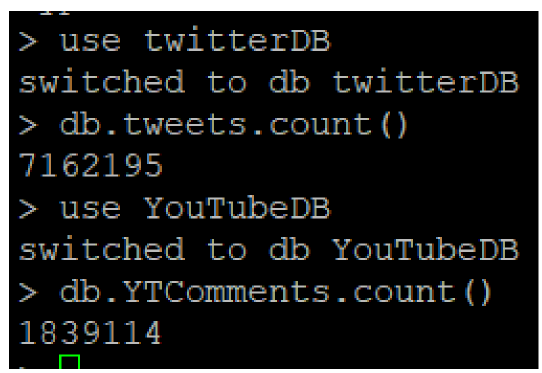

Figure 5 shows the number of Tweets and Comments that are stored in MongoDB

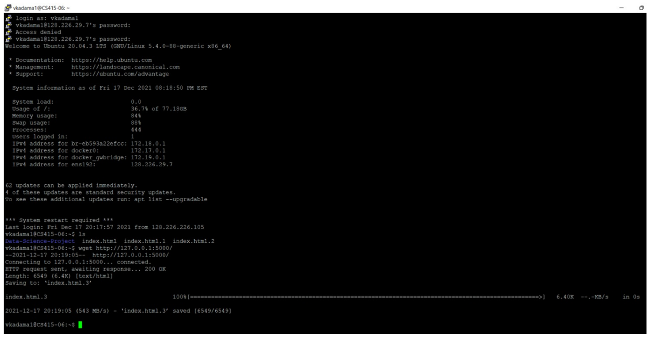

Figure 6a Executing the Flask Script

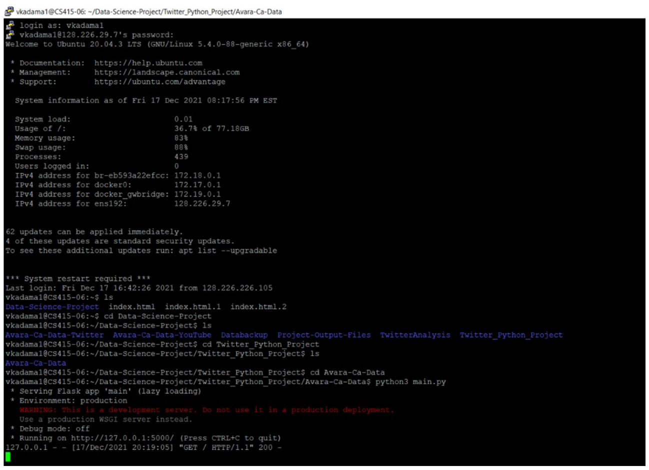

Figure 6b Executing the Flask Script

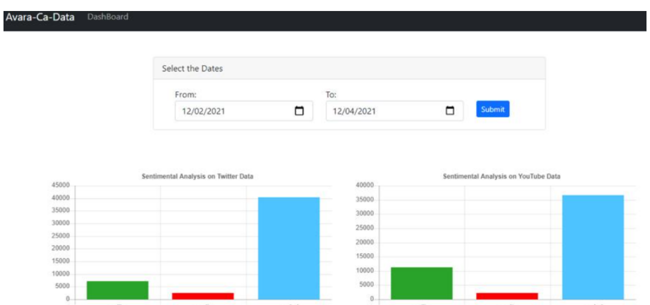

Figure 7 Flask WebApp Homepage

10 OBSERVATIONS

Figure 2 shows the total no of tweets vs the tweets and total YouTube comments vs useful comments. We found that all the tweets and comments we collected are useful for the analysis.

Figure 2a shows the number of tweets collected per day. X-axis represents the dates and Y-axis represents the number of Tweets collected.

Figure 2b shows the number of YouTube comments collected per day. X-axis represents the dates and Y-axis represents the number of YT Comments.

Figure 4 shows that the size of DB after collecting all the data from Twitter and Youtube, for storing Youtube comments the size is 0.132GB and for twitter it is 1.105GB.

Figure 5 shows that we have collected and then stored 7162195 (Approximately 7.1 Million) Tweets and 1839115 (Approximately 1.8 Million) Youtube Comments in MongoDB.

11 PROJECT 3 IMPLEMENTAION

For the dashboard, we have developed a webapp using python flask and html, which takes two dates as input in YYYY-MM-DD format and plots the requested graphs between those two dates. When we execute the flask python script, as seen in figure 6a and 6b, the webapp is hosted on our VM's localhost:5000 port, so We did SSH tunneling (Port forwarding) to access the web API from the local browser and now we can input the dates and generate graphs there. When we input the dates and click the submit button, in the back-end, a function is called with the two dates as input parameters, and the function generates two bar graph plots in the front end, one with Sentimental analysis on Twitter data and other with Sentimental analysis on YouTube data. We then analyze the sentiments on tweets and YouTube comments that we fetch in the time period and return the number of positive, negative and neutral sentiments to the flask app.

12 DASHBOARD

As we can see in the Figure 7, it represents the home page of our flask webapp, where we can enter the dates in YYYY/MM/DD format to generate those two graphs. The range for the dates we can select should be in between 2021/11/02 to 2021/11/12 as we have our data collected on that specific time period. Now user can select the dates within the range described above to generate sentimental analysis of both tweets collected from twitter and YouTube comments.

13 BUG REPORTS

We do not have any major bugs, but sometimes based on our selected date range on the dashboard the plots take some time to generate, since they are pulling data in real time from Mongo database, and also the tweets and YouTube comments volume is huge.

14 RESULTS

Final plots for YouTube and twitter were generated and from Figure 7 we can see the Number of tweets in the Y-axis for twitter graph and in the X-axis we can see the sentiments classified based on our sentiment analysis . Similarly, We can see the Number of YouTube comments in the Y-axis for the other graph and in the X-axis we can see the sentiments classified based on our sentiment analysis.

15 REFERENCES

Rawan Fahad Alhujaili and Wael M.S. Yafooz(2021). Sentiment Analysis for Youtube Videos with User Comments: Review.

Albert Biffet and Eibe Frank(2010). Sentiment Knowledge Discovery in Twitter Streaming Data. Discovery Science, Lecture Notes in Computer Science, Volume 6332/2010, 1-15, DOI: 10.1007/978-3-642-16184-1

Alec Go, Richa Bhayani and Lei Huang(2009). Twitter Sentiment Classification using Distant Supervision. Project Technical Report, Stanford University.

Mohd Majid Akhtar(2019). Sentiment Analysis on Youtube Comments: A brief study. Jamia Milia Islamia

Pinkesh Badjatiya1(2017), Shashank Gupta(2017), , Manish Gupta(2017), Vasudeva Varma(2017). Deep Learning for Hate Speech Detection in Tweets. IIIT-H, Hyderabad, India

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, Rebecca PassonneauSentiment. Analysis of Twitter Data. Department of Computer Science. Columbia University.