

Twitter Sentiment Analysis for Brand Improvement

Parth D Patel¹, Yash R Patel², Kishan Patel³, Sharvil Patel⁴

¹parth.p2@ahduni.edu.in

²yash.p3@ahduni.edu.in

³kishan.p1@ahduni.edu.in

⁴sharvil.p@ahduni.edu.in

^{1,2,3,4}Information and Communication Technology

School of Engineering and Applied Science, Ahmedabad University

Ahmedabad, India.

Abstract— This project is an aim to create a model for companies and organizations that can deliver better experiences to their customers across the world on the basis of thoughts, beliefs, and mindset of customers themselves through their tweets tweeted on the world-famous microblogging platform Twitter which entertains around 6000 tweets per second. Opinion of people matters a lot to analyze how the propagation of information impacts the lives in a large-scale network like Twitter. Sentiment Analysis of the tweets decides the polarity and inclination of a vast population towards a specific topic, item, person or entity. The ultimate goal is to analyze twitter data to perform algorithms and predict better results through Machine Learning models to evaluate a company or brand in real-time on the basis of the past datasets as well as live data streaming and for changes in their business models, detect anomalies with alert and for enforcing public engagement.

Keywords— Twitter Sentiment Analysis | Feature Extraction | Classification | Trend Analysis | Regular Expression | Naive Bayes Classification | Data Visualization |

I. INTRODUCTION

Today, the era of the Internet has changed the lifestyle and the way people interact with each other and express their views or opinions on it and which is mainly done through blog posts, review websites, social media etc. Millions of people are using social networking sites like Facebook, Twitter, LinkedIn, Google Plus, etc. to express their emotions, and share their views of their daily lives as well as current trending topics (such as a COVID pandemic) going in the world. Through these online platforms or communities, we get interactive data where consumers inform and influence others using different forums. In this manner, social media generates a huge amount of data in the form of blogs, tweets or comments on posts and reviews etc which ultimately leads to sentiments (or semantics). Social media also provides a platform for advertising to customers and generates business opportunities through campaigning and various ads which is also a great emerging field for online income generation. People mostly depend upon this type of user generated content for decision making because of the fact that they are maximum influenced to that in comparison to any other source, especially where now you can see the whole world news in a piece of mobile phone in your hand. For e.g., if someone wants to purchase an item or needs to utilize any service, at that point they initially look for the reviews and discussions on the

web, examine it via online media forum, discuss, and talk before making a choice of actually acquiring the product or service.

This project addresses the sentiment analysis in Twitter; i.e. classifying tweets according to the sentiment expressed in them: positive, negative or neutral. Analysing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socio-economic phenomena like stock exchange and then making relevant changes in their business models and strategies. The aim of this project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream (obtained from Twitter API) and letting the brands make it easier to know about the trends going on in the real market and making decisions through the results obtained from it.

II. LITERATURE SURVEY

Through Sentimental Analysis, Businesses wish to extract point of view from the online reviews (here tweets) in order to improve their products and in turn their reputation/growth and help in customer satisfaction. Sentiment analysis is learning of people's emotions, views, attitude, and opinions. It is also known as opinion mining. Sentiment analysis identifies the sentiment articulated in a text and then analyzes it. Sentiment Analysis is a type of Natural Language Processing (NLP) technique for tracking the mood of the public through a particular product but more it is a super set of it containing the NLP technique. There is a wide range of applications of Sentiment Analysis in real life usage and one such we will be describing here in detail along with the process, models, and inferences.

Twitter Sentiment Analysis is a very known topic in this field due to its wide range of applications in various fields such as research, politics, trend recognition, personality tracking, brand improvement, etc. Through a wide range of research in this field from past few years, it has obtained a huge amount of attention for its application in this field and a lot of articles and projects have been developed. The scale to which this can be implemented is now reaching new heights and we aim to do one such research project involving the techniques of machine learning that can be useful for real life applications at a large scale for companies, brands, individuals as well as governments.

Sentiment Analysis can be used in many applications in various fields like,

1. Applications that use Reviews from Websites:

As in today's Internet world, there has been a large collection of reviews and feedback on almost everything. This includes product reviews, feedback on political issues, comments about services of any specific company or brand etc. So to maintain any company's business goal, it is a need for a sentiment analysis system that can extract sentiments about a particular product or services. It will assist us in optimising the process of providing input or ratings for a given product, item given by any user over social media (eg. twitter). This would serve the needs of both the users and the companies/organizations.

2. Applications in Business Intelligence

In today's era, it has been observed that people nowadays tend to look upon reviews of products which are available online before they buy them. So for the majority of businesses/brands, the online opinion decides the success or failure of their product. Thus, Sentiment Analysis plays an important role in businesses. Marketing field companies use it to develop their strategies, to understand customers' feelings towards products or brands, how people respond to their campaigns or product launches and why consumers don't buy some products.

By tracking public views, important data regarding sales, trends and customer satisfaction can be extracted. Sentiment Analysis can also be used in trend prediction which is what we commonly see trending in hashtags section on Twitter.

III. IMPLEMENTATION

The implementation of Twitter Sentiment Analysis is done by utilizing the APIs by Twitter itself. First, we understand the basic sentiments of the customer/public (tweet which are considered in our case) which a brand needs to monitor and take action on.

Here is the whole approach to perform sentiment analysis sentiments from tweets as follows:

1. Download twitter testing data from the twitter api and add it in the database. Here, what we do is called directly collecting streaming data from the Twitter API with the help of Tweepy, MySQL and Python.
2. For this purpose, we need to apply and create a developer account on Twitter Developers. This is the access point for obtaining Twitter API's and other various tools related to this analysis regarding Twitter tweets.
3. Afterwards, we create an app there to generate two API keys and two access tokens to call the Twitter API which is unique for every user associated with his/her account. These are saved and copied carefully since available only one time.
4. Then, we use streamer and listener for oauth and connect the Tweepy with our unique access and token

keys and finally connect our local MySQL database to it. We also check for the on_status method for using different objects in the preprocessing part such as text, polarity, subjectivity, geo object, etc.

5. Next in preprocessing is cleaning the tweets by removing the stop words as well as removing emoji characters. Then, we tokenize each word from the dataset and analyze it in the program.
6. For each token or word, we compare it with the positive sentiments and negative sentiments words predefined in the list and increase the respective frequency counts for it.
7. Finally, from the final sentiment score we can decide its polarity like positive tweet (+1), Negative tweet (-1) or Neutral tweet (0) and analyze the results through it by using data visualization tools such as Dash and Plotly as well as host it on a cloud ready server such as Heroku.

The figure 1 shows the sentimental analysis algorithm as per the approach.

Algorithm 1 Extract Twitter sentiment

```

1: procedure TWITTER-CONNECTION()
2:   consumer - key = '#####'
3:   consumer - secret = '#####'
4:   access - token = '#####'
5:   access - token - secret = '#####'
6:   self.auth = OAuthHandler(consumer-key, consumer-secret)
7:   self.auth.set_access_token(access - token, access - token - secret)
8:   self.api = tweepy.API(self.auth)
9: end procedure
10:
11: procedure TWEET-CLEANING(t)
12:   tweet = t.remove - Stop-words
13:   Return tweet
14: end procedure
15:
16: procedure TWEET-CLASSIFICATION(t)
17:   t = Tweet - Cleaning(t)
18:   tweet - polarity = t.sentiment.polarity
19:   Return tweet-polarity
20: end procedure
21:
22: procedure GET-TWEETS(q, count)
23:   fetched-tweets = self.api.search(q = query, count = count)
24:   Return fetched - tweets
25: end procedure
26:
27: procedure MAIN()
28:   st = SentimentalTwitter()
29:   tweets = st.fetch - tweets(query = 'politics', count = 300)
30:   PositiveTweets = tweets that sentiment = 'positive'
31:   NegativeTweets = tweets that sentiment = 'negative'
32:   NeutralTweets = tweets that sentiment = 'Neutral'
33:
34:   for tweet t in PositiveTweets do
35:     print(t)
36:   end for
37:   for tweet t in NegativeTweets do
38:     print(t)
39:   end for
40:   for tweet t in NeutralTweets do
41:     print(t)
42:   end for
43: end procedure

```

Figure 1: Extract Sentiment from Tweets

Initially, we have used the sample data of relatively small dataset of tweets for trying our initial machine learning model. Then, after exploring the Twitter Developer account and its relevant resources, we found out about the use case of Twitter API and found the required tweets data regarding different brands and companies from different locations (states and countries). All the steps are mentioned below.

We have used python to implement the sentiment analysis and the packages that we have utilized are tweepy and textblob. We installed the required libraries by following commands:

- 1) pip install tweepy
- 2) pip install textblob

The python library textblob is used for text processing. It uses the NLTK package for NLP (Natural Language Processing) which is typically here used for text processing techniques like cleaning, tokenizing, and removing stopwords, removing emojis, etc.

The Connection to Twitter API is done using MySQL on a local server and to connect we need to create an account on twitter and define an application. We can generate the api keys from this web link [Twitter Developers](#). The Application settings are shown below in figure 2.

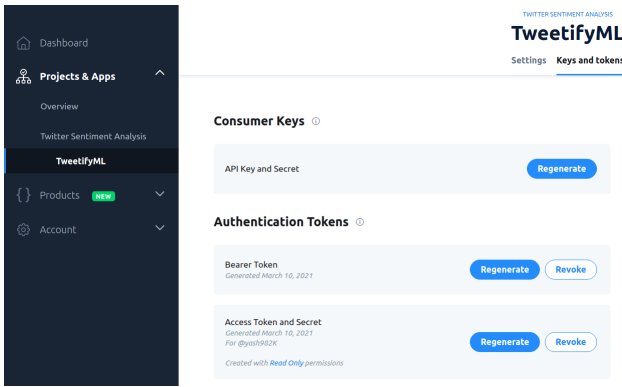


Figure 2: Twitter Application Management Console

In the intermediate part, we need to pre-process the data, and to make it model application ready, It involves various factors, like use of special characters such as “@” and “#” as well as emojis in tweets which are very popular. We need to check about other factors too such as stopwords, POS tagging, noun phrase extraction, spelling correction, parsing and use of multiple n-gram models for efficient deployment and application of model with an aim to achieve as maximum accuracy as possible.

When we extract the tweets we check for mentions, hashtags for our brand which we are tracking. Also, if available, locations and geographical segmentation factors are taken into consideration for more details of the dataset and then generating trends and results as per it. Then using those tweets which extract the features of that tweet (tokenization and data cleaning). Then, using the trained model we use a bag of

predefined words or lexicons. Then using that model we classify the test data (live tweets).

For Classification Algorithm, We have used the Naive Bayes Classifier algorithm. It is a probabilistic classifier and it can learn the pattern of examining a set of documents that has been categorized.^[10] It compares the words with the predefined content of the list and classifies them to the right category. Let t be the tweet and c^* be a class assigned to t , where

$$C^* = \arg \max_c P_{NB}(c | t)$$

$$P_{NB}(c | t) = \frac{(P(c)) \sum_{i=1}^m p(f_i | c)^{n_{i(t)}}}{P(t)}$$

From the above equation, ‘ f ’ is a feature, count of feature (f_i) is denoted as $n_{i(t)}$ and it is present in t which represents a tweet. Here, m denotes no. of features. Parameters $P(c)$ and $P(f_i | c)$ are computed through maximum likelihood estimates. To train and classify using Naive Bayes Machine Learning technique, we can use the Python NLTK library.

IV. RESULTS

This project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet for users around the world. Finally, we plot a pie chart to show the percentage of each sentiment. For further better visual representation, we aim to deploy the live streaming of dataset through applications, tools and technologies such as Plotly. Here run time data of live tweets can be fetched from the official data of Twitter API.

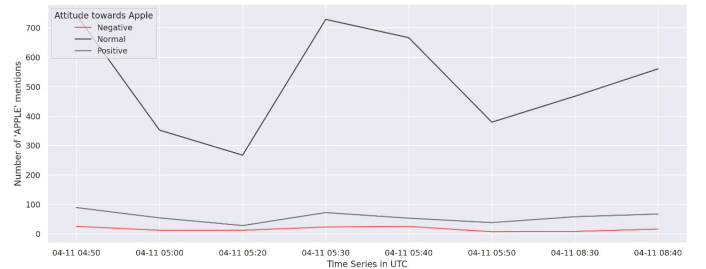


Figure 3: Attribute towards Apple

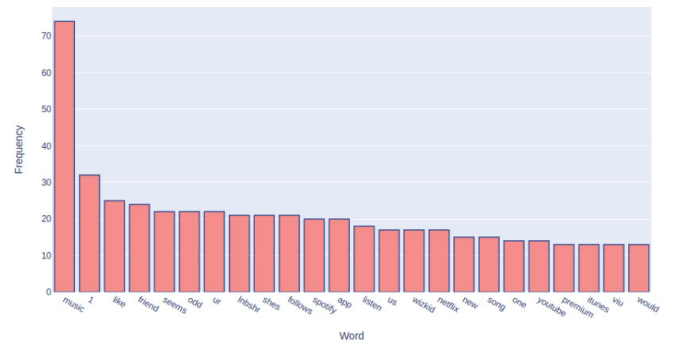


Figure 4: Frequency of Tokens from Tweets

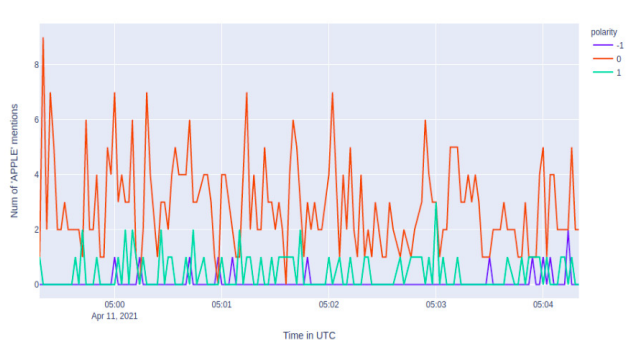
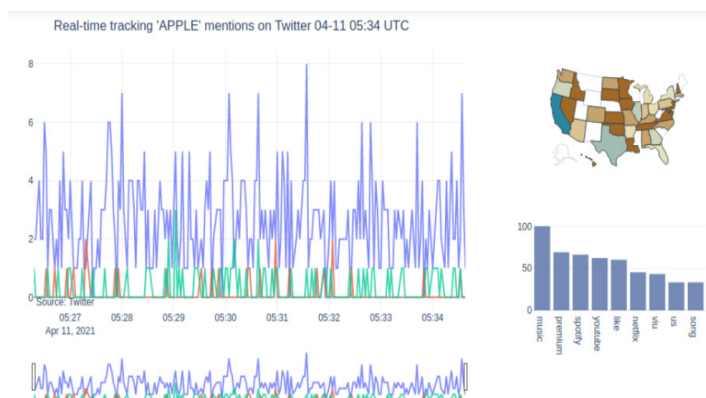


Figure 5:

The above graphs illustrate the output of the project. The first graph here shows the tweets being classified in three norms:- positive, negative or neutral. The company here we tracked is 'APPLE'. And, the time zone which we tracked is UTC. The values show in the next graph are +1 for positive, 0 for neutral and -1 for negative. This graph is a very interactive way to analyze the data and it shows the real-time activity of each time a tweet is captured.

The final part of the graph shown above is the amalgamation of all the individual plots on a single page. This is right now showing the live tracking of the tweets based on different polarities as well as geometrical segmentation. The graphs can be maximised and minimized as per the user's choice and they can be evaluated well for the companies and organization to analyze. This, as a conclusion determines the ratio of the feelings or views about any product for a company which is mentioned in the tweets by the users either in form of text, emojis or hashtags. Some of the works which can be still implemented are parallelizing streams and

V. CONCLUSION



In this paper about sentence-level sentiment analysis, users are getting information from the online platform whether it may be a positive polarity or negative polarity or neutral. Finally they come under specific polarity from the reviews or views or opinions from the online like social media. Sentences may text

or emotions or emoticons or expressions. Even though emotions look like a diagram, it represents some textual information which are very well represented through the analysis done from algorithms, models and. Although Twitter messages have unique characteristics compared to other corpora, machine learning algorithms are shown to classify tweet sentiment with similar performance. We also studied the effects of various features on classifiers. At last we can conclude that more the cleaner data, more accurate results can be obtained.

VI. REFERENCES

- 1] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," *Stanford.edu*. [Online]. Available: <https://cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf>. [Accessed: 17-Mar-2021].
- 2] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," *Aclweb.org*. [Online]. Available: <https://www.aclweb.org/anthology/P11-1016.pdf>. [Accessed: 17-Mar-2021].
- 3] I. G. Council, R. Mc Donald, L. V. Google, and 76 Ninth Avenue, "What's great and what's not: Learning to classify the scope of negation for improved sentiment analysis," *Googleusercontent.com*. [Online]. Available: <http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/36744.pdf>. [Accessed: 17-Mar-2021].
- 4] "Twitter Sentiment Analysis using Python - GeeksforGeeks," *Geeksforgeeks.org*, 24-Jan-2017. [Online]. Available: <https://www.geeksforgeeks.org/twitter-sentiment-analysis-using-python/>. [Accessed: 17-Mar-2021].
- 5] "For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool," *Sentiment140.com*. [Online]. Available: <http://help.sentiment140.com/for-students>. [Accessed: 17-Mar-2021].
- 6] A. J. Schouby, "Descriptions, truth value intuitions, and questions," *Linguist. Philos.*, vol. 32, no. 6, pp. 583–617, 2009.
- 7] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends® Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.

- 8] M. Afham, "Twitter Sentiment Analysis using NLTK, Python," *Towards Data Science*, 25-Sep-2019. [Online]. Available: <https://towardsdatascience.com/twitter-sentiment-analysis-classification-using-nltk-python-fa912578614c>. [Accessed: 17-Mar-2021].
- 9] *Iitk.ac.in*. [Online]. Available: https://cse.iitk.ac.in/users/cs365/2015/_submissions/ajaysi/slides.pdf. [Accessed: 17-Mar-2021].
- 10] Pablo Gamallo, Marcos Garcia, "Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets", 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, Aug 23-24 2014, pp 171-175.
- 11] H. Bagheri and M. J. Islam, "Sentiment analysis of twitter data," *Arxiv.org*. [Online]. Available: <https://arxiv.org/ftp/arxiv/papers/1711/1711.10377.pdf>. [Accessed: 11-Apr-2021].
- 12] Y. Patel, [CSE523-Machine-Learning-TweetifyML](#).