

UNDERSTANDING THE IMPACT OF SENTIMENT OF TWEETS IN A SOCIAL NETWORK GRAPH

[Document subtitle]



Team Members:

Devila Bakhrania

Abhishek Singh

Aditya Shenoy

Aditya Sonawane

Heramb Dharamadhikari

Vinay Balani

Contents

[Abstract 2](#_Toc56959467)

[Introduction 2](#_Toc56959468)

[Data Extraction & Exploration 2](#_Toc56959469)

[Data Cleaning 5](#_Toc56959470)

[Analysis 6](#_Toc56959471)

[Natural Language Processing 6](#_Toc56959472)

[Sentiment Analysis 7](#_Toc56959473)

[The Training and Prediction 8](#_Toc56959474)

[Feature Extraction from Text 8](#_Toc56959475)

[Classification Algorithms 8](#_Toc56959476)

[Network Graph 8](#_Toc56959477)

[Conclusion 12](#_Toc56959478)

# Abstract

In the time of crisis of COVID-19 when every human was made to socially-distance themselves from each other for their own safety and to curb the spread of the virus, the world went into quarantine, some took “social distancing” with a sigh of relief and some on the other hand as sigh of frustration. Such emotion spread like a forest fire in the social network giving rise to a chain reaction. We are planning to analyze the rate of spread and the impact it causes which could be positive or negative.

# Introduction

Big data is exactly what it sounds like—a lot of data. Alone, a single point of information cannot provide you much understanding. But terabytes of information, combined beside complex scientific models and tumultuous computing control, can make experiences human creatures are not able of creating.

In today’s uncertain and unpredicted times where human fears contact is a result of direct impact of social media and interactions within it. Topics such as Racism, LGBTQ, COVID-19, and many more have become sensitive and are being used to contexts and ways that are entirely wrong in terms human interaction. To curb such inhuman practices, we must analyze the root cause and classify them as negative perception to create awareness.

Objectives:

* Classify the tweet content into positive and negative baskets
* Mapping tweets and differentiating them using colors or other methodologies based on followership
* Demonstrating the rate and extent of type of tweets at which these create impact
* Finding the best possible way to promote or create awareness about a sensitive topic such as COVID-19 precautions

# Data Extraction & Exploration

The first step to big data analytics is gathering the data itself. This is known as “data mining.” Data can come from anywhere. Most businesses deal with gigabytes of user, product, and location data. In this tutorial, we will be exploring how we can use data mining techniques to gather Twitter data, which can be more useful than you might think.

Twitter is a gold mine of data. Unlike other social platforms, almost every user’s tweets are completely public and pullable. This is a huge plus if you are trying to get a large amount of data to run analytics on. Twitter data is also specific. Twitter’s API allows you to do complex queries like pulling every tweet about a certain topic within the last twenty minutes or pull a certain user’s non-retweeted tweets.

A simple application of this could be analyzing how your company is received in the public. You could collect the last 2,000 tweets that mention your company (or any term you like) and run a sentiment analysis algorithm over it.

We also target users that specifically live in a certain location, which is known as spatial data. Another application of this could be to map the areas on the globe where your company has been mentioned the most.

We created a Developer Account to extract the huge amounts of tweets analyze for more accurate insights. Few of the commands which were crucial to the entirety of the project are as follows:

* **API.search(*q*[, *geocode*][, *lang*][, *locale*][, *result\_type*][, *count*][, *until*][, *since\_id*][, *max\_id*][, *include\_entities*])**

Returns a collection of relevant Tweets matching a specified query.

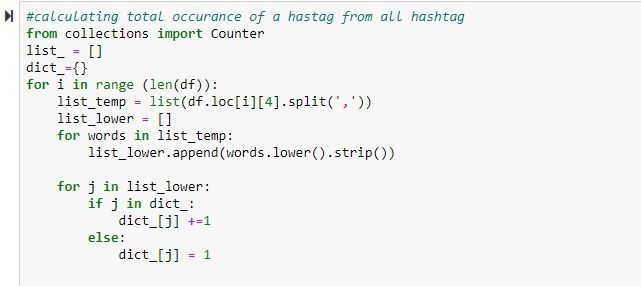
* **API.retweeters(*id*[, *cursor*][, *stringify\_ids*])**

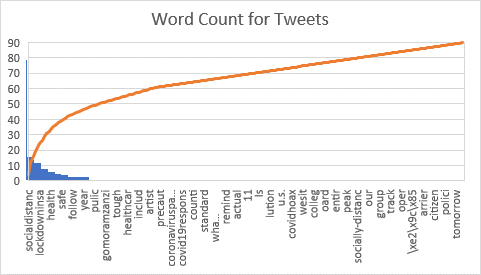
Returns up to 100 user IDs belonging to users who have retweeted the Tweet specified by the id parameter.

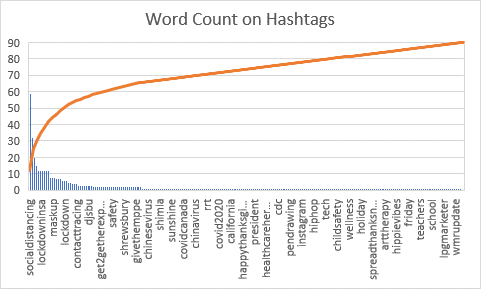
* **API.followers([*id/screen\_name/user\_id*][, *cursor*])**

Returns a user’s followers ordered in which they were added. If no user is specified by id/screen name, it defaults to the authenticated user.

We started with extracting the top 100 used words and the corresponding count to find the keyword that has highest impact. We see “socialdistancing” has the highest frequency. Hence, we decided to analyze the sentiment, positive or negative, associated with each tweet.







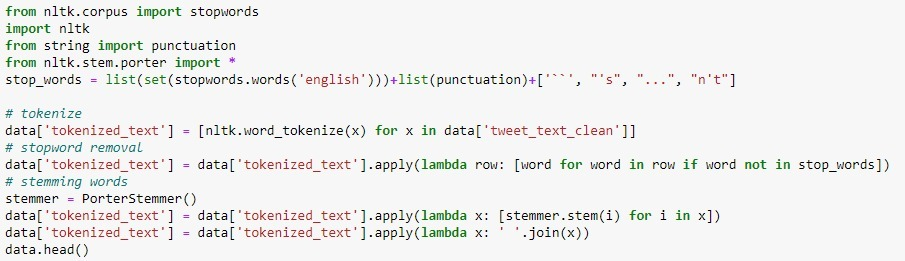
The Pareto’s line in the above 2 graphs shows suggests that the cumulative count of the words flattens as we approach the less frequently used words. This means the 80% of the data on twitter is from the top 20% of the most frequently used words.



The “tweet.user.location.encode('utf-8')” attribute in tweepy.Cursor object provides us with the access to geo location of choice.

# Data Cleaning

The data extraction process does not stop here. This data must be cleaned by removing stop words, line breaks, and garbage out to perform an accurate analysis. The data for a specific location, with specific tweet content and authenticated users was contain over 5000 users. Upon cleaning this data set we got a fine-tuned data of 3500 users.



To perform analysis on a balanced dataset we filtered 80 rows with equal distribution over 5 geo locations, namely Manhattan, Brooklyn, Queens, Bronx, and Staten Island.

# Analysis

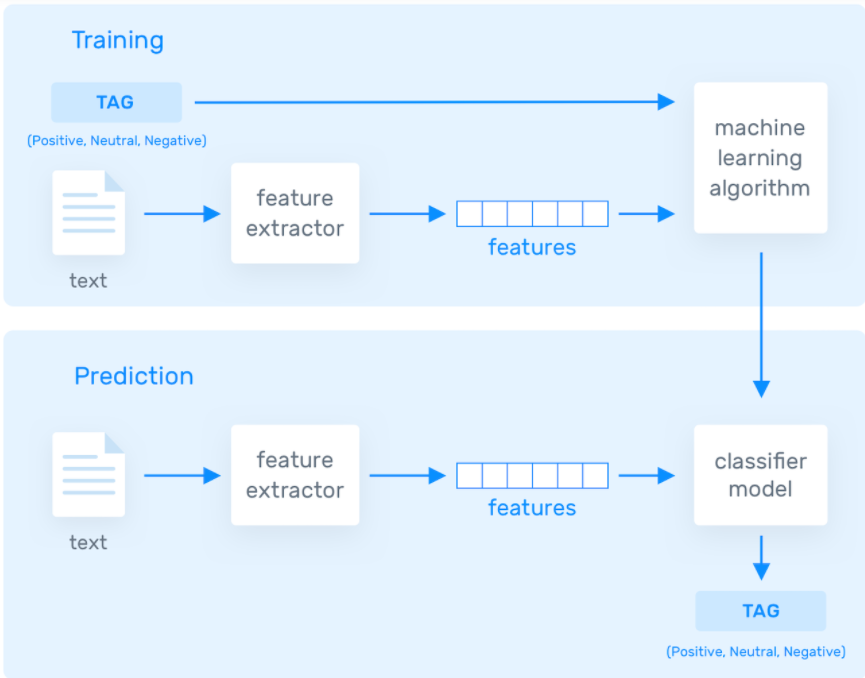
## Natural Language Processing

Natural Language Processing (NLP) permits machines to break down and translate human language. It is at the center of devices we utilize each day – from interpretation computer program, chatbots, spam channels, and look motors, to language structure redress computer program, voice collaborators, and social media checking apparatuses.

Natural Language Processing (NLP) is a field of Artificial Intelligence (AI) that makes human language intelligible to machines. NLP combines the power of linguistics and computer science to study the rules and structure of language and create intelligent systems (run on machine learning and NLP algorithms) capable of understanding, analyzing, and extracting meaning from text and speech.

Natural language understanding (NLU) is used to understand the structure and meaning of human language by analyzing different aspects like syntax, semantics, pragmatics, and morphology. Then, computer science transforms this linguistic knowledge into rule-based, machine learning algorithms that can solve specific problems and perform desired tasks.

Utilizing content vectorization, NLP instruments change content into something a machine can get it, at that point machine learning calculations are bolstered preparing information and anticipated yields (labels) to prepare machines to create affiliations between a specific input and its comparing yield. Machines at that point utilize measurable investigation strategies to construct their possess “knowledge bank” and perceive which highlights best speak to the writings, some time recently making forecasts for inconspicuous information



At the end, more data is fed to train the algorithm, more accurate results are acquired.

## Sentiment Analysis

We have utilized the automatic approach of sentiment analysis. Automatic methods, contrary to rule-based systems, do not depend on manually made rules but on machine learning methods. An opinion investigation assignment is as a rule modeled as a classification issue, whereby a classifier is encouraged a content and returns a category, e.g. positive, negative, or impartial.

### The Training and Prediction

In the training process, our model learns to associate a particular input (i.e. a text) to the corresponding output (tag) based on the test samples used for training. The feature extractor transfers the text input into a feature vector. Pairs of feature vectors and tags (e.g. positive, negative, or neutral) are fed into the machine learning algorithm to generate a model.

In the prediction process, the feature extractor is used to transform unseen text inputs into feature vectors. These feature vectors are then fed into the model, which generates predicted tags (again, positive, negative, or neutral).

### Feature Extraction from Text

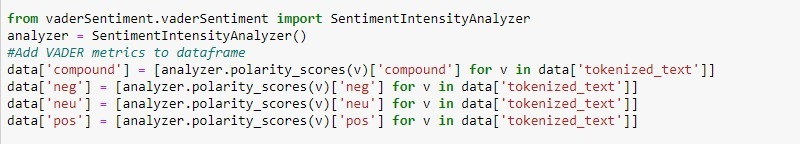
The first step in a machine learning text classifier is to transform the text extraction or text vectorization, and the classical approach has been bag-of-words or bag-of-ngrams with their frequency.

More recently, new feature extraction techniques have been applied based on word embeddings (also known as word vectors). This kind of representations makes it possible for words with similar meaning to have a similar representation, which can improve the performance of classifiers.

### Classification Algorithms

The simplest solutions are usually the most powerful ones, and Naive Bayes is the algorithm we trust will be most helpful. Despite the incredible propels of machine learning, it has demonstrated to not as it were basic but too quick, exact, and solid. It has been effectively utilized for numerous purposes, but it works especially well with normal dialect handling (NLP) issues.

Naive Bayes is a family of probabilistic algorithms that take advantage of probability theory and Bayes’ Theorem to predict the tag of a text (like a piece of news or a customer review). They are probabilistic, which means that they calculate the probability of each tag for a given text, and then output the tag with the highest one. The way they get these probabilities is by using Bayes’ Theorem, which describes the probability of a feature, based on prior knowledge of conditions that might be related to that feature.

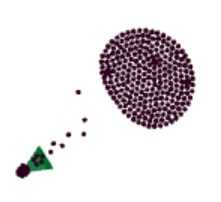


# Network Graph

Once we had the entire data clean and analyzed, we go to the next step of plotting the network graph. The attributes we used for plotting are as follows:

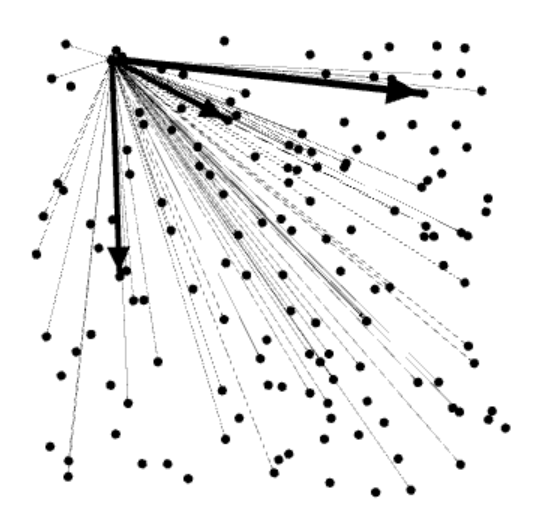
* Timestamp
* Tweet\_text
* Username
* Retweet\_count
* Sentiment
* Relative normalized weight

The weight of each edge is calculated by the relative change from the source to target node.

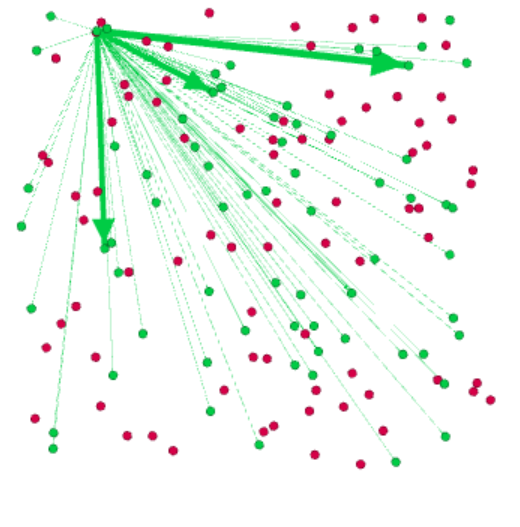


We have plotted a section of the entire social network so to observe the directed edges more clearly and accurately. The above Network Graph plot is as we see in the above image.

Here is the first draft of the Social network graph without sentiment analysis implementation.



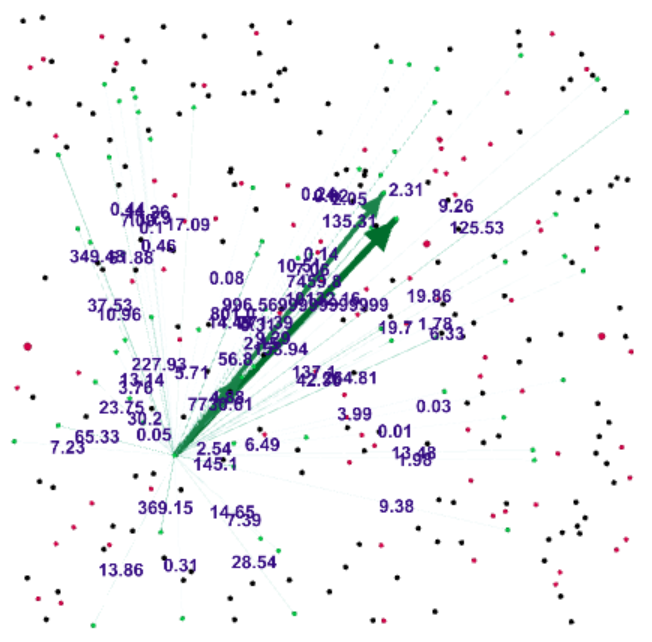
The sentiment is given as an added component to the graph to differentiate between the Positive(green) and Negative(red) nodes.



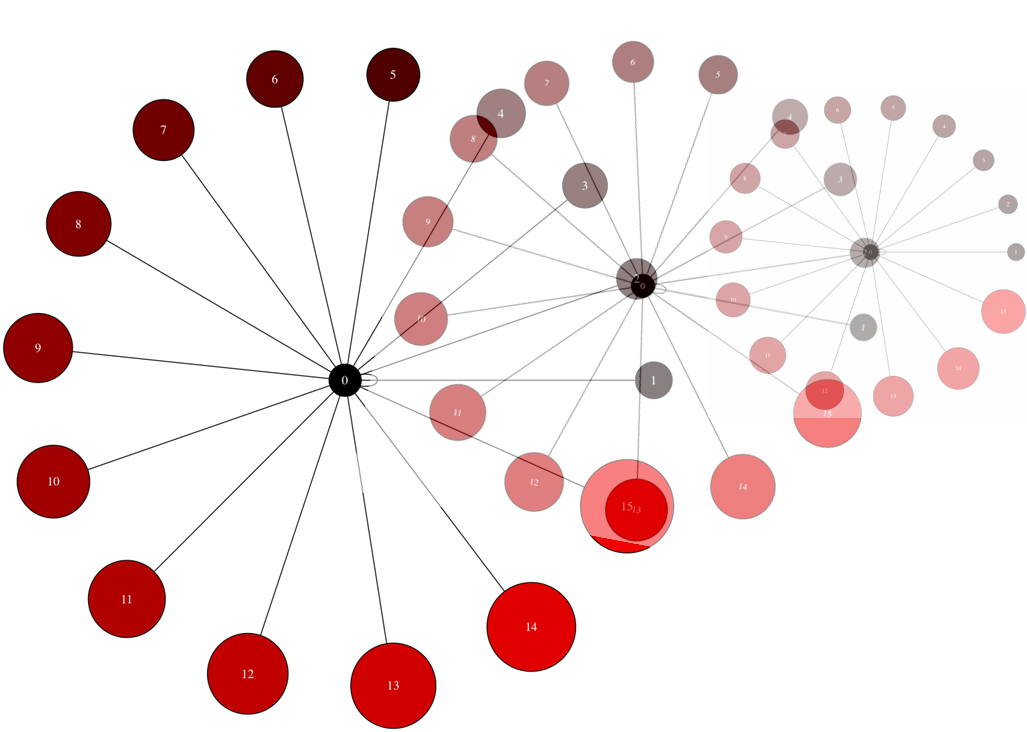
The are labelled according to the username so as to differentiate between them. Additionally, we can use the username to traceback to the origin of a sentiment spread.



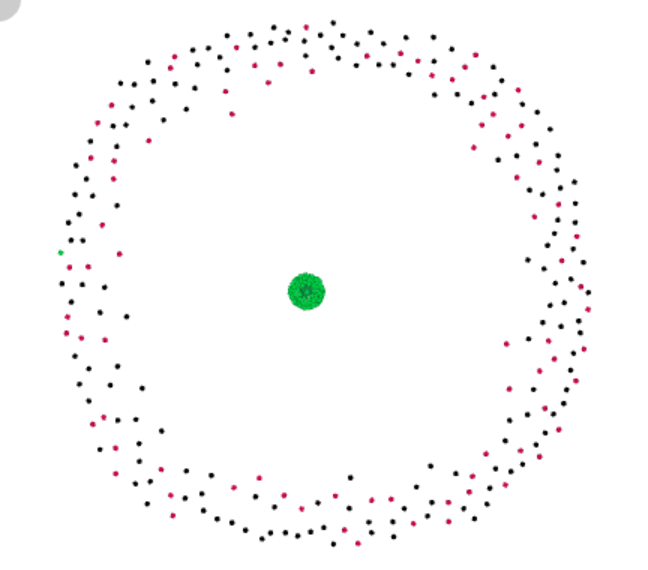
A few edge weights have need filtered to get a better visual insight out of the graph.



A tru social network graph is complex and messy as every node and subnode has its on group of subnodes which makes it impossible to visualize. Here is a demonstration



To make it more visually insightful, we have plotted 2 layers of the social network from the source node of a tweet.



# Conclusion

In this project, we have demonstrated how a tweet whether sending out a good message or a not so good on, has an impact how it spreads across the social network. Each negative tweet does not necessarily get converted into a negative tweet after one hop and vice versa.

Additionally, we have also explored the impact of the positive impact spreads at a much faster rate than a negative spread as we see much more proportion of green nodes in a balanced and unbiased dataset.

One more feature we that is worth exploring is whether the information about relative position of word in a tweet has any effect on the performance of the classifier.

Last but not the least, we can attempt to model human confidence in our system. If we implement the same project on a larger dataset, we can traceback a crucial part of the shortest path to find the origin of a sentiment thereby understanding what are the main components of affecting the specific phenomenon.