

## Goals and Interesting Findings

In creating this project, I wanted to investigate the things that people tend to talk positively and negatively about over social media. Twitter is one of the largest platforms that individuals often share their interests, feelings, and actions. With that in mind, the goal of this study was to detect themes found in positive and negative Twitter posts and ultimately build a model that can classify a post as having either positive or negative sentiment.

The initial takeaways from this study were the themes that were found in positive and negative Twitter posts. After performing some descriptive analytics, I found that positive Tweets are often praising things associated with weekends and entertainment. Conversely, negative Tweets were often talking about everyday tasks and worries as well as technology. The other major takeaway from this study was that the sentiment of a tweet can be effectively classified using a logistic regression model where the significance of different words in determining the sentiment of a tweet is deduced by using the term frequency-inverse document frequency technique.

## Data Collection and Preparation

The source of all of the data related to this study was a Kaggle dataset containing 1.6 million Twitter posts from 2009<sup>1</sup>. Of the 1.6 million Tweets, 800,000 were Tweets with positive sentiment and 800,000 were Tweets with negative sentiment. The dataset had all emoticons removed from the contents of the Twitter posts and labeled the Tweets as having either a positive or negative sentiment. Given that this dataset had already labeled the sentiment for each Tweet, this dataset was ideal for developing a model to classify the sentiment of a string of text. Additionally, the large number of entries in the dataset allowed me to get a good idea of the things that people were talking positively and negatively about.

Since the dataset already contained all of this information, the only actions necessary for making the most of the dataset were cleaning the Tweets to make them more useful in my analysis. This involved removing numbers and special characters, removing usernames where posts mentioned other users, and removing words shorter than three characters as shorter words tend to be less meaningful in determining the sentiment of a string of text. I also removed the suffixes from all words in the dataset. This ensures that words like happy, happier, happiest, and happiness are all treated the same in my analysis. With a robust, cleaned dataset, I could move on to analyzing the data.

## Trends in Twitter Posts

The first insight that I wanted to draw from the dataset was to see if people had a tendency to post positive Tweets about certain ideas and negative Tweets about other ideas. To do this, I performed descriptive analytics, looking at the most frequently occurring words in Tweets with positive sentiment and Tweets with a negative sentiment. Figures 1 and 2 portray the most common words in the form of a Word Cloud. In the Word Cloud, larger words depict words that appeared most frequently in the dataset.

---

<sup>1</sup> <https://www.kaggle.com/kazanova/sentiment140>

[illegible]

Looking at Tweets with a positive sentiment, I found that several words related to weekend activities like drinking, sleeping, and partying. I also found that many positive Tweets contained words related to forms of entertainment like movies, television shows, and music. This makes sense as these are all activities and that occur frequently and are not limited to a specific time of year like holidays or special events.

## Classification Model

In addition to understanding the things that people were often praising and complaining about, I wanted to build a model that would classify Tweets as either positive or negative. To do this, I used the Tweets and their labeled sentiment from the dataset to train a logistic regression model that predicts the sentiment of a string of text. Since the only input for the model to make a prediction from is a string of text, I had to use a method to extract the significance of different words in the dataset so that the model can make predictions on word groupings that it has never seen before. To do this, I used a method called term frequency-inverse document frequency. This

method considers how many times each word appears in a Tweet and compares it to the number of times that word appears in all of the rest of the Tweets. That way, you can extract which words are important in determining the outcome of the Tweet versus words that are common in all Tweets. I have built a Jupyter notebook that walks through the process of fitting the model on a small scale<sup>2</sup>.

After training the model, I calculated the F1 score of the model to assess its accuracy on validation data to fine-tune my model, as well as on test data to see how the model performed on unseen data. The F1 score is a metric that assesses the accuracy of a classification model like the one I built by considering the odds that a predicted positive result is a true positive and the odds of the model detecting a true positive. A higher F1 score indicates the model fits the data better. For the validation data, the model's final F1 score was approximately .75, while the F1 score for the model on the unseen test data was roughly .55. This decrease is to be expected as the model altered to optimize performance on the validation data but not the test data, and therefore, the F1 score for the test data is an unbiased estimate of the performance of the final model. The alterations performed to optimize the model's performance on the validation data included changing the minimum number of characters that a word must have to be considered in the model and changing the predicted probability that would signify the model's prediction of a Tweet with positive sentiment

Ultimately, by extracting the significance of different words in influencing the sentiment of a string of text using term frequency-inverse document frequency, I was able to train a logistic model that was reasonably successful in classifying strings of text as either positive or negative.

## Dashboard Application

After ensuring that the model was working as it should and making reasonably good predictions on unseen data, I could build an application that would generalize and apply the model to any string of text that a user could enter. When building the application, since I knew that the model was working sufficiently well when being trained on only part of the dataset, I decided to train the final model on all 1.6 million tweets in the dataset before applying it in the dashboard.

Now that the model had been trained on all of the entries in the dataset, the dashboard could be built to make predictions on user-entered text. To do this, the text had to be transformed into a term frequency-inverse document frequency matrix like the data that the model had been trained on, to determine the impact of the words in the text. The model then predicts the probability that the text has a positive sentiment. If that probability is greater than .5, the application tells the user that the text is positive. If the probability is less than .5, it tells the user that the text is negative. Additionally, the dashboard displays all of the words contained in the text that was entered. A word is displayed in green if it was found more in the positive Tweets, a word is displayed in red if it was found more in the negative Tweets, and a word is displayed in black if it does not appear in the dataset. Although this is not a perfect representation of how words are influencing the model's prediction, it can give a good estimate of positive and negative words since there is an equal number of positive and negative Tweets in the training data.

---

<sup>2</sup> <https://deepnote.com/project/fa91b211-2975-412c-a8b0-839a93b97fa6#%2FVisualizationAndModelFitting.ipynb>

In addition to a string of text, the dashboard also allows the user to enter the URL to a specific Tweet and makes a prediction on the contents of the Tweet. To do this, I extracted the post identification number from the URL and used the Twitter API to scrape the contents of the Tweet. The text from the Tweet is then converted to a term frequency-inverse document frequency matrix and a prediction is made in the same way as if a user enters a string of text.

The application that uses the final model can be seen at the website <https://nlp-twitter-sentiment.herokuapp.com/> and the GitHub repository containing all of the code for every part of this study can be found at <https://github.com/patgeitner/MA346TwitterSentimentAnalysis>.

## Summary

After performing descriptive analytics by looking at the most frequently occurring words in Tweets with both positive and negative sentiment, some patterns arose that allowed me to draw conclusions about what people tend to praise and complain about on social media. First, by grouping words like “sleeping”, “drinking”, and “party” I observed that people often look forward to and praise weekends and social events. Additionally, I was able to cluster words like “movie”, “watch”, “listen”, and “song” to conclude that people often post positive things around entertainment and media.

In looking at patterns that arose in Tweets with negative sentiment, I grouped words like “work”, “school”, “test”, and “morning”. This allowed me to observe that people often complain about the tasks and challenges that they face daily. I also found that negative Tweets were often centered around technology through the frequency of words like “iPhone”, “phone”, and “update”.

Looking at the patterns found in positive and negative Tweets side-by-side, we can see that people often complain about the foundational aspects of everyday life. Activities like work and school take up a majority of people’s time so it makes sense that people would have the most negative things to say about them. On the other hand, we can see that the things that people tend to praise in their social media posts are things that help them escape everyday activities. Whether it is through parties, sleeping, listening to music, or watching movies, these are all times where people feel more removed from their everyday life.

Additionally, using the term frequency-inverse document frequency technique to extract information about words and how they contribute to the sentiment of a string of text, I was able to train a model that can make reasonably good predictions about whether a text is positive or negative. However, this model could be improved by addressing some limitations. First of all, the dataset used to train the model was from 2009. Since then, there are certainly words whose meanings and use cases have been slightly altered, especially in social media posts. This could be addressed by training the model on multiple years of data. Additionally, the model does not consider the impact of punctuation, however, I feel that in some cases, punctuation can certainly impact the sentiment of a string of text. It would be interesting to conduct a future study on the impact of punctuation in determining the sentiment of a text.