Project 3 Group 18

**Group Members:** Maiah Jaffa @jaffamaiah, Andres Portillo @andresporcruz, Martin Wall @wall-martin

**GitHub Repository:** https://github.com/Andresporcruz/social-media-sentiment-analysis.git

**Video Link:**

**Problem Statement** (rewrite)

In this project, we developed a sentiment analysis model that quantifies the level of sentiment in an input statement, based on social media data. This project works with data specific to public opinions on various topics, events, and entities. Through training a model with social media data from text-based platforms like X we quantify input statements as either positive or negative while also quantifying the approximate level of sentiment of a statement as a percentage.

**Motivation Statement** (rewrite)

Society has shifted in recent years, and consumers and constituents can share their opinions on a variety of topics with ease. It is vital that organizations take advantage of the opportunity to analyze this data and make informed decisions that will benefit those they serve. Sentiment analysis insights can be useful in gauging public opinion on products, services, or policies. Further this work can aid organizations in identifying and addressing customer complaints promptly, monitoring and responding to PR crises effectively, and understanding trends and shifts in public perception over time.

**Features Statement** (rewrite)

This project allows for users to input a statement and receive a score of 1 or 0, with 1 indicating positive sentiment and 0 indicating negative sentiment based on our simple algorithm. The user can also toggle to the multinomial algorithm to calculate a phrase's sentiment percentage, where a higher percentage indicates a higher sentiment value. Overall, this project yields 75% accuracy for the sentiment analysis model {can it recall above 80% for each sentiment category}.

**Data**

The Sentiment140 dataset, sourced from Kaggle, contains 1.6 million tweets, each labeled with a sentiment score. The schema includes the following fields: Tweet\_ID (int), a unique identifier for each tweet; Timestamp (datetime), the date and time of the tweet; Text (string), the content of the tweet; and Sentiment (int), the sentiment label, with -1 for negative, 0 for neutral, and 1 for positive.

**Tools**

**Programming Language:** C++

SFML: For creating an interactive dashboard.

Cstdio: For passing input to and receiving output from python scripts

**Programming Language:** Python

NLTK: For natural language processing tasks.

scikit-learn: For implementing machine learning algorithms.

pandas: For data manipulation and analysis.

matplotlib and seaborn: For data visualization.

**Algorithms implemented:**

Additional Data Structures/Algorithms used:

Distribution of Responsibility and Roles: Who did what?:

Any changes the group made after the proposal? The rationale behind the changes.

Initially we were interested in classifying statements into three categories: positive, neutral, and negative. Providing users with a more nuanced understanding of sentiment in statements through capturing and classifying statements that aren’t inherently sentimental or not sentimental. However, in progressing through our project we prioritized reaching an accuracy rate as close to 85% accurate as possible. In development we found that in our efforts to achieve a higher accuracy rate it would be best to work with just two classifications. The ambiguity of classifying neutral statements led to a much more complex model with decreased performance.

We also added a second algorithm to analyze the statements. Th multinomial statement we added gives users a further insight into the sentiment value of their input statement. {Add rationale/justification}. Additionally, we initially planned on providing the user with data visualizations so they could explore sentiment data. However, we wanted to focus on showing the users the sentiment values of their statements as opposed to the sentiment values of the model data.

Big O worst case time complexity analysis of the major functions/features you implemented

As a group, how was the overall experience for the project?

Did you have any challenges? If so, describe.

If you were to start once again as a group, any changes you would make to the project and/or workflow?

Comment on what each of the members learned through this process.

References