**Project Report: Sentiment Analysis Tool for Twitter**

* **Sumit Kumar**
* **Tharun Prabhakar**
* **Niranjan Kumar Kishore Kumar**

**Introduction:**

Twitter aims to develop a sentiment analysis tool to help its clients understand the benefits of their products based on Twitter data. This report outlines the development and evaluation of a sentiment analysis model to address this need.

**Stakeholder and Problem Statement:**

**Stakeholder:** Twitter

**Problem Statement:** Develop a sentiment analysis tool to identify and analyze sentiments towards client products based on Twitter data, providing insights into public perception and potential areas for improvement.

**About Dataset:**

The dataset consists of tweets related to various companies, and it is primarily aimed at sentiment analysis. Below is a detailed description of the dataset:

* **Number of Entries:** 74,682 rows
* **Columns:**
  + 1. **Tweeet\_Id:** Identifier for tweets (integer).
    2. **Company:** The name of the company mentioned in the tweet (string).
    3. **Label:** Sentiment label of the tweet, which includes categories like Positive, Negative, Neutral, etc. (string).
    4. **Tweet:** The text of the tweet (string).
* **Data Quality and Missing Values:**

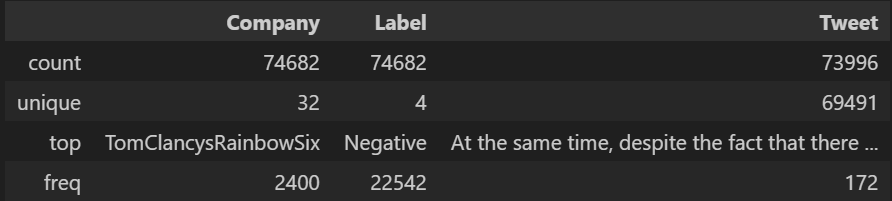
**Missing Values:** The 'Tweet' column has some missing entries, with 73,996 non-null entries out of 74,682, suggesting that a small number of tweets are either empty or not recorded.

* **Distribution of Sentiment Labels:**
  + 1. **Sentiment Categories:** The dataset includes multiple sentiment categories, likely including Positive, Negative,

Irrelevant and Neutral.

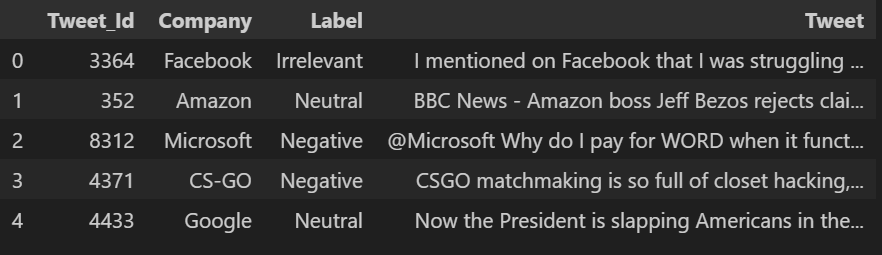
* + 1. **Distribution:** The sentiment labels are distributed across the dataset, with 'Negative' being the most frequent category.
    2. **Company Mentions:**

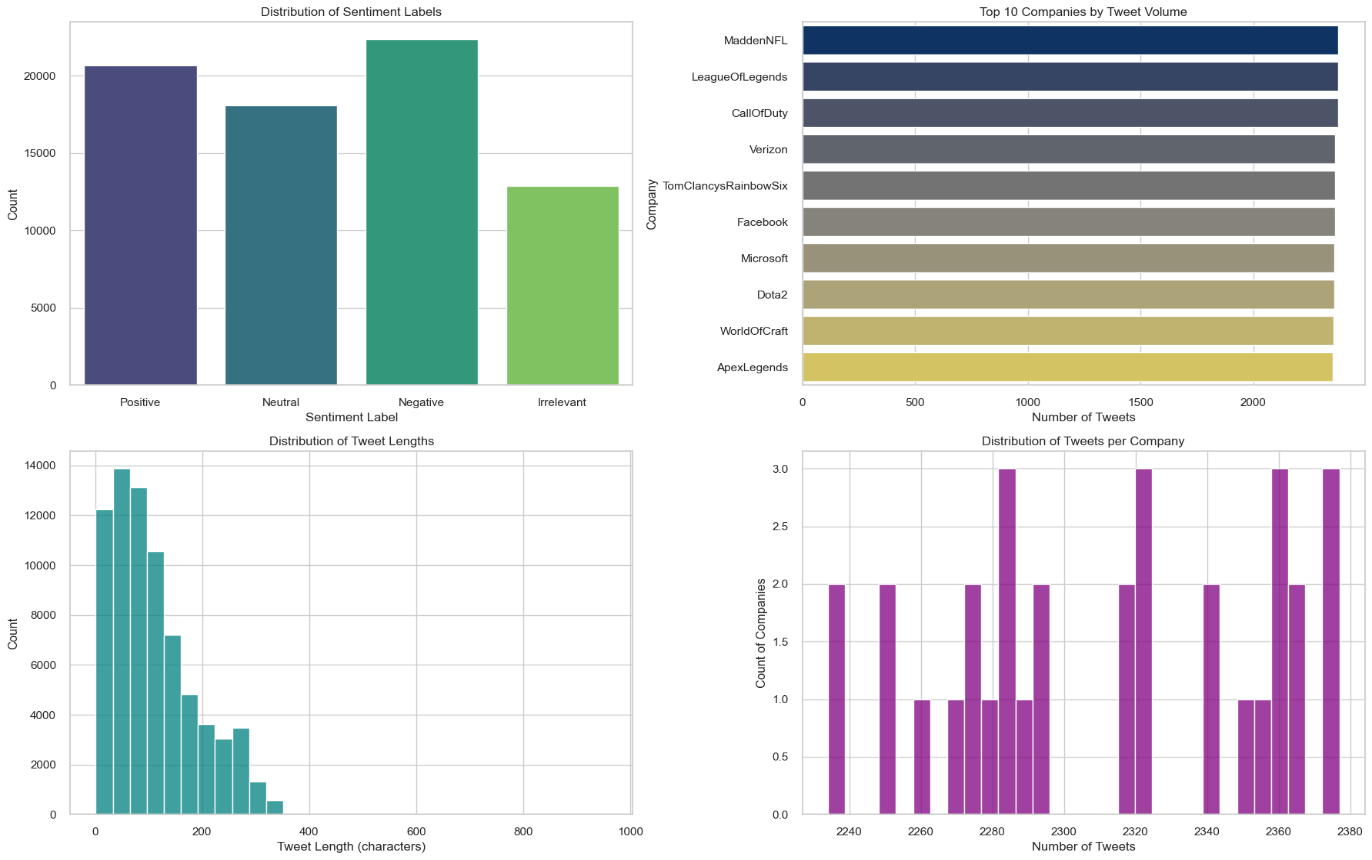
**Unique Companies:** There are 32 unique companies mentioned across the tweets.

**Tweet Volume:** Some companies are mentioned more frequently than others, with 'TomClancysRainbowSix' being one of the most mentioned.

* **Tweet Characteristics:**

1. **Length:** The length of tweets varies, with most tweets being relatively short, typical of Twitter's character limit constraints.
2. **Unique Tweets:** The 'Tweet' column contains 69,491 unique entries out of 73,996 non-null tweets, indicating some level of duplication or similar tweets.

* **Example Data:**

**Exploratory Data Analysis:**

* **Distribution of Sentiment Labels:**
  1. **Positive:** There are around 21,000 tweets labeled as positive.
  2. **Neutral:** Approximately 17,000 tweets are labeled as neutral.
  3. **Negative**: Around 23,000 tweets are labeled as negative, making it the most frequent sentiment label.
  4. **Irrelevant:** There are about 13,000 tweets labeled as irrelevant.

The dataset has a balanced distribution of sentiment labels, although negative sentiments are slightly more prevalent. This balance is crucial for training a sentiment analysis model to ensure it performs well across all categories.

* **Top 10 Companies by Tweet Volume**

1. **Top Companies:** The companies with the highest number of tweets include MaddenNFL, LeagueOfLegends, CallOfDuty, Verizon, TomClancysRainbowSix, Facebook, Microsoft, Dota2, WorldOfCraft, and ApexLegends.
2. **Volume:** Each of these companies has around 2000 tweets mentioning them, with MaddenNFL having the highest volume.

These companies are likely to be the focus of sentiment analysis due to their high volume of mentions. The high tweet volume indicates significant public interest and engagement, making them important subjects for understanding public sentiment.

* **Distribution of Tweet Lengths:**

1. **Tweet Lengths:** Most tweets are relatively short, with a high frequency of tweets having lengths between 0 to 200 characters. The distribution shows a right skew, with fewer tweets having lengths beyond 200 characters.

The majority of tweets are concise, aligning with Twitter's character limit. This insight is important for preprocessing, indicating that feature extraction methods should be optimized for shorter texts.

* **Distribution of Tweets per Company:**

1. **Tweet Counts per Company:** The number of tweets per company shows a somewhat uniform distribution around certain counts, with most companies having around 2240 to 2380 tweets.

The distribution suggests that the dataset includes a consistent number of tweets for each company. This consistency can help in performing comparative sentiment analysis across different companies without significant bias due to tweet volume disparity.

* **Word cloud:**

****

The word cloud visualization provides a visual representation of the most frequent words in the dataset. Here are some insights from the word cloud:

* + 1. **Prominent Words:**

**"game":** This is one of the most prominent words, indicating a strong focus on gaming-related content in the tweets.

**"twitter":** The frequent mention of "twitter" suggests that many tweets are discussing the platform itself, which is expected given the source of the data.

**"play":** This word's prominence aligns with the focus on gaming, indicating that many tweets are about playing games.

**"unk":** This might be a placeholder or a result of data preprocessing. It may need further investigation.

**"pic":** Commonly used in tweets to refer to pictures or images, indicating that visual content is often mentioned.

* + 1. **Other Frequent Words:**

**"new":** Suggests discussions about new releases or updates, which could be new games, features, or products.

**"love":** Indicates positive sentiment towards certain subjects, possibly games or features.

**"s\*\*t":** A frequent negative term, which might indicate complaints or negative sentiments.

**"now", "one", "will", "play", "still":** Commonly used words that might be part of frequent phrases or discussions.

* + 1. **Brand and Product Mentions:**

**"Facebook", "Verizon", "Microsoft":** These company names appear in the word cloud, suggesting that the tweets frequently mention these brands.

**"Red Dead", "GTA", "Fortnite", "PS5":** Specific games and gaming consoles are also mentioned, highlighting popular topics in the dataset.

* + 1. **Emotional and Reaction Words:**

**"love", "good", "great", "fun":** Positive words indicating favorable reactions.

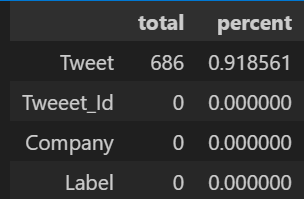
**"s\*\*t", "f\*\*k":** Negative words indicating complaints or frustrations.

**"thank":** Indicates gratitude, possibly in response to customer service or product satisfaction.

**Data Pre-processing:**

* **Handling Missing Values:**

We can see that there are 686 missing values. Since the number of null tweets are less compared to the dataset, they were removed.

****

* **Lowercasing Letters:**

Ensures uniformity by converting all text to lowercase, which helps in reducing the variability of the text data. For example, "Game" and "game" will be treated as the same word.

* **Remove URLs, Hashtags, Mentions, and Special Characters:**

Removes unwanted noise such as URLs, hashtags, and mentions, which are not useful for sentiment analysis. This step also removes special characters to keep only alphanumeric characters and spaces.

* **Remove Numbers:**

Eliminates numbers from the text, which may not be relevant for sentiment analysis. For example, specific dates or numbers in tweets usually do not contribute to the sentiment.

* **Tokenize Text:**

Splits the text into individual words (tokens), which is a crucial step for further processing such as removing stopwords and lemmatization.

* **Remove Stopwords:**

Removes common words (stopwords) that do not contribute much to the sentiment or meaning of the text, such as "and", "the", "is", etc. This helps in focusing on the more meaningful words.

* **Lemmatization:**

Reduces words to their base or root form (lemma). For example, "running" becomes "run". This helps in standardizing words and reducing the dimensionality of the text data.

* **Join Words to Reconstruct the Text:**

Reconstructs the processed words back into a single string. This step is necessary if the subsequent processing or modeling steps expect text input rather than a list of tokens.

* **Vectorization:**

When dealing with text data for sentiment analysis, two common vectorization methods are **CountVectorizer** and **TF-IDF Vectorizer**. **CountVectorizer** transforms text into a matrix of token counts, making it simple and easy to interpret. It is suitable for datasets where word frequency is important, but it can lead to high-dimensional, sparse matrices. **TF-IDF Vectorizer**, on the other hand, weighs tokens by their importance using Term Frequency-Inverse Document Frequency. This method reduces the impact of common words and highlights significant ones, often leading to better performance for complex datasets. TF-IDF is ideal for longer texts or when the importance of words in context is crucial. Both methods can produce high-dimensional data, but TF-IDF typically offers more meaningful representations, especially for varied and large vocabularies. Experimenting with both can help determine the best fit for your specific needs.

**Model Tried:**

We tried several models for sentiment analysis, including:

* + - Decision Tree
    - Random Forest
    - Gradient Boosting
    - AdaBoost
    - Multinomial Naive Bayes

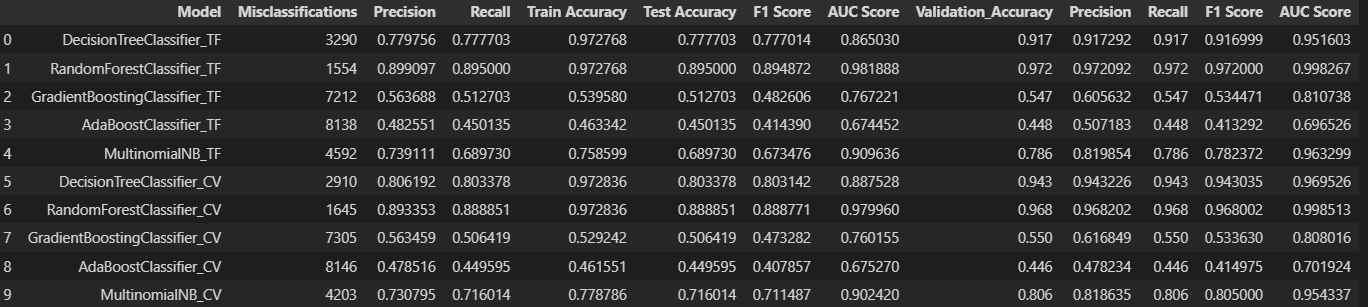
We chose these models because they are commonly used for text classification tasks like sentiment analysis. Each model has its strengths and weaknesses, and experimenting with multiple models allows us to find the one that best suits our dataset and problem.

**Evaluation Metric:**

We evaluated the models using various metrics to assess their performance. These metrics include:

* **Accuracy:** To measure the overall correctness of the model's predictions.
* **Precision**: To measure the proportion of true positive predictions among all positive predictions, focusing on the model's ability to avoid false positives.
* **Recall**: To measure the proportion of true positive predictions among all actual positive instances, focusing on the model's ability to capture all positive instances.
* **F1** **Score**: To balance precision and recall, providing a single metric that considers both false positives and false negatives.
* **AUC** **Score**: To measure the model's ability to discriminate between positive and negative classes, particularly useful for imbalanced datasets.

We chose these metrics because they provide a comprehensive understanding of the model's performance from different perspectives, considering both correct and incorrect predictions.



**Future Work:**

1. Given more time, we would explore more advanced techniques for text preprocessing and feature engineering, such as word embeddings or deep learning-based approaches.
2. We would also experiment with ensemble techniques like stacking or blending to further improve model performance.
3. Additionally, we would conduct more extensive hyperparameter tuning to optimize each model's performance.
4. Exploring domain-specific lexicons or sentiment dictionaries could also enhance the model's understanding of sentiment in the context of specific industries or products.

**Client Recommendation:**

1. Based on the evaluation results, we recommend using the Random Forest Classifier using Count vectorizer for sentiment analysis of Twitter data.
2. The precision and recall achieved by the model are good enough for the intended use case. However, the specific thresholds for precision and recall may depend on the client's requirements and the consequences of false positives and false negatives in their decision-making process.

**Model Deployment:**

1. We can deploy the trained model as a web service or API using platforms like Flask or Django.
2. The model can also be integrated into existing systems or applications using libraries like TensorFlow Serving or FastAPI.
3. Deployment considerations include scalability, latency, security, and monitoring to ensure the model performs well in production environments.