WEEK 6

Progress Report

# Introduction

Now that we have prepared the Twitter Friends dataset available on Kaggle, the next step is to develop a model that can accurately identify the sentiment of tweets posted by Twitter users based on their followers and friends count. The goal of this model is to classify tweets as positive, negative, or neutral based on the language used and the overall sentiment expressed.

To accomplish this, we will be utilizing various natural language processing techniques such as text preprocessing, feature extraction, and sentiment analysis. Our aim is to create a model that can effectively analyze the language used in tweets and accurately classify them into the appropriate sentiment category.

Through this project, we hope to contribute to the field of sentiment analysis and provide a useful tool for analyzing social media data.

# Problem Definition:

The task is to develop a machine learning model that can analyze tweets posted by Twitter users and classify them into positive, negative, or neutral sentiments based on the language used and the overall sentiment expressed. The model should use the Twitter Friends dataset available on Kaggle, which contains information about the number of followers and friends for each Twitter user, as a feature for sentiment classification. The goal is to create a model that can accurately classify tweets based on their sentiment, which can be useful in a variety of applications, such as market research, customer feedback analysis, and brand reputation management

# Steps to achieve

1. Data Preparation: Start by loading and cleaning the Twitter Friends dataset. This may involve removing duplicates, missing values, and irrelevant columns, as well as performing any necessary text preprocessing steps, such as lowercasing, tokenization, stop word removal, and stemming or lemmatization.
2. Data Exploration and Visualization: Once we have preprocessed the data, we can explore and visualize it to gain insights into the distribution of sentiments, the most frequent words or phrases, and the correlation between different features and the sentiment label.
3. Feature Extraction: The next step is to extract relevant features from the text data to feed into the machine learning model. This can be done using different techniques such as bag-of-words, tf-idf, word embeddings, or deep learning architectures such as Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN).
4. Model Selection: Once we have extracted the features, we can train and evaluate different machine learning models to classify the sentiment of tweets as positive, negative, or neutral. Some popular models for sentiment analysis include Naive Bayes, Logistic Regression, Support Vector Machines (SVM), Decision Trees, Random Forests, and Gradient Boosting.
5. Model Evaluation and Hyperparameter Tuning: After training the models, we should evaluate their performance using metrics such as accuracy, precision, recall, and F1-score. We can also perform hyperparameter tuning using techniques such as grid search, random search, or Bayesian optimization to optimize the model's hyperparameters and improve its performance.
6. Deployment and Integration: Once we have selected the best performing model, we can deploy it to a web or mobile application, or integrate it with other systems for real-time sentiment analysis of Twitter data.

# Suitable models

1. **Naive Bayes Classifier:** This is a probabilistic model that is simple, fast, and easy to implement. It assumes that each feature (word) is independent of the other features, which may not always be true. However, it can still perform well for sentiment analysis tasks.
2. **Logistic Regression:** This is a linear model that can be used for binary or multi-class classification tasks. It works well for sentiment analysis when the feature space is high-dimensional and sparse.
3. **Support Vector Machines (SVM**): This is a powerful algorithm for classification tasks that can handle high-dimensional and non-linear data. SVM can perform well for sentiment analysis tasks, especially when combined with kernel tricks.
4. **Decision Trees:** This is a non-parametric model that can handle non-linear data and is easy to interpret. However, it may suffer from overfitting if the tree is too complex.
5. **Random Forests:** This is an ensemble learning technique that combines multiple decision trees to improve the accuracy and reduce overfitting. It can perform well for sentiment analysis tasks and is also robust to noisy data.
6. **Gradient Boosting:** This is another ensemble learning technique that combines multiple weak classifiers to create a strong classifier. It can also perform well for sentiment analysis tasks and is especially good at handling imbalanced datasets.

# Next week projection

Next week, we could focus on feature extraction from the Twitter Friends dataset using techniques such as bag-of-words, tf-idf, and word embeddings. Once we have extracted the relevant features, we can start training and evaluating different machine learning models such as Naive Bayes, Logistic Regression, and Support Vector Machines (SVM) for sentiment classification.

We can then evaluate the performance of these models using metrics such as accuracy, precision, recall, and F1-score. Based on the evaluation results, we can select the best performing model and perform hyperparameter tuning to further improve its performance.

Finally, we can deploy the model to a web or mobile application for real-time sentiment analysis of Twitter data.

# Conclusion

In conclusion, developing a model for sentiment analysis on Twitter data can provide valuable insights into how people perceive a particular topic, brand, or event. In this project, we will use the Twitter Friends dataset available on Kaggle and follow the standard steps of feature extraction, model selection, evaluation, and deployment to develop an accurate and efficient sentiment analysis model. By implementing different machine learning algorithms and techniques, we can compare their performance and select the best-performing model for real-time sentiment analysis of Twitter data. This project can have applications in various industries, including marketing, finance, politics, and social media analysis.