[Applied Analytics Project Week 5 Assignment]

Major: Applied Analytics

Name: Heng Bai, Yuang Guo

Data processing:

This week, we processed a Twitter sentiment dataset with columns such as 'id', 'company', 'sentiment', and 'text' by applying several key preprocessing techniques to prepare it for machine learning tasks. Feature engineering started with tokenization, splitting each text entry into individual tokens to transform the raw textual data into a format suitable for algorithms. We applied stemming and lemmatization to reduce words to their base forms (e.g., "running" to "run"), ensuring that different word forms were treated consistently. The text was also converted to lowercase to avoid case sensitivity issues, and we used label encoding to transform categorical sentiment labels ('Positive', 'Negative', 'Neutral') into numerical values for easier interpretation by the models. In addition, we applied text embedding techniques, including Count Vectorizer, which converted the text into a matrix of token counts; TF-IDF, which assessed word importance based on frequency; and Word2Vec, which generated dense, lower-dimensional vectors to capture semantic relationships between words, making the data more structured for analysis.

For data augmentation, we introduced various methods to increase the dataset's size and diversity. Synonym replacement using WordNet was applied to substitute words with their synonyms (e.g., "good" replaced by "great"), creating alternative versions of the text without altering the meaning. We also used random word insertion and word swapping to introduce variations by adding new words or rearranging existing ones. Noise injection, involving the addition of misspelled words or slight modifications, helped simulate real-world inconsistencies in the text data. These augmentation methods enhanced the model's robustness by ensuring it could generalize to unseen data and avoid overfitting.

Lastly, we performed dimensionality reduction using Word2Vec embeddings, which transformed each word into lower-dimensional vectors while preserving semantic context. This step significantly reduced the complexity of the dataset, helping to streamline the model training process without sacrificing important information. High-dimensional data can lead to inefficiencies and overfitting, so reducing the feature space was crucial for model performance. These preprocessing steps—feature engineering, data augmentation, and dimensionality reduction—ensured that the dataset was optimized for machine learning models, enhancing both interpretability and computational efficiency.