A Capstone Project report submitted

in partial fulfillment of requirement for the award of degree

**BACHELOR OF TECHNOLOGY**

in

**SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE**

by

**2203A52110 SIDDHARTHA NAMILIKONDA**

Under the guidance of

**Dr.Ramesh Dadi**

Assistant Professor, School of CS&AI.

SR UNIVERSITY

SR University, Ananthsagar,Warangal,Telagnana-506371

**CONTENTS**

# DATASET

## *Project-1: The Air Quality*

**The Air Quality dataset** contains detailed information about atmospheric pollution levels and associated meteorological conditions recorded at regular intervals. It includes measurements of key pollutants such as PM2.5, PM10, $NO_2$, CO, $SO_2$, and $O_3$, along with environmental parameters like temperature, humidity, wind speed, and wind direction. Each entry is timestamped and tagged with location-specific data, allowing for temporal and spatial analysis of air pollution. This dataset is widely used for studying pollution trends, forecasting air quality, and assessing environmental and health impacts in urban and industrial region

## *Project-2: Plant disease detection Classification*

**The Tomato Bacterial Spot dataset** is used for detecting and classifying bacterial spot disease in tomato leaves through image-based analysis. It consists of images of tomato plant leaves, both healthy and affected by bacterial spot, a common disease caused by *Xanthomonas* bacteria. The disease typically appears as small, dark, water-soaked spots on the leaves, which can enlarge and lead to leaf yellowing, defoliation, and reduced crop yield. This dataset supports the development of machine learning and deep learning models aimed at early detection and classification of the disease, enabling timely intervention and improved crop management practices.

## *Project-3: The Sentiment Analysis Text Dataset*

**The Sentiment Analysis Text Dataset** consists of a large collection of text samples, such as movie reviews, tweets, product feedback, or customer comments, each labeled with a sentiment category—typically positive, negative, or neutral. These datasets are used to train machine learning and deep learning models to automatically detect and interpret emotions or opinions expressed in text. By analyzing linguistic patterns, word associations, and contextual clues, models built on sentiment analysis datasets can provide valuable insights into public opinion, brand perception, and user satisfaction across various domains.

# METHODOLOGY

## Project 1: The Air Quality dataset analysis

**Data Collection and Preprocessing:**

The project utilized an air quality dataset comprising various atmospheric pollutant indicators recorded across India. The dataset was imported into a Pandas DataFrame and subjected to an initial cleaning process. Non-essential columns such as 'Date' and 'Time' were removed to streamline the analysis. A preliminary check for missing values was conducted. Columns with excessive missing data were dropped, and the remaining missing values were imputed using the median strategy via SimpleImputer. Additionally, invalid or non-numeric entries were identified and corrected to maintain data integrity.

**Feature Engineering and Outlier Removal:**

Numerical features were isolated to facilitate statistical analysis. Histograms were plotted to examine the distribution of each feature, while boxplots were used to detect the presence of outliers. The Z-score method was applied to filter out extreme values (i.e., those with a Z-score greater than 3), ensuring that the data used for model training was representative and free from distortion due to anomalies.

**Exploratory Data Analysis (EDA):**

Exploratory analysis was conducted to uncover underlying patterns and relationships between different pollutant variables. Scatter plots were used to observe pairwise relationships, while skewness and kurtosis were computed to assess the distribution shape of each feature. This analysis helped identify non-normal distributions, which could potentially impact model assumptions and performance.

**Model Training:**

Three regression models were selected for evaluation: Linear Regression, Random Forest Regressor, and XGBoost Regressor. The dataset was split into training and testing subsets to evaluate model generalization. Standardization of features was performed where appropriate using StandardScaler to enhance model training and convergence behavior.

**Performance Evaluation:**

Model performance was assessed using standard regression metrics—Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and $R^2$ score. The models were compared based on these metrics to determine their effectiveness in predicting pollutant concentrations. Additionally, the influence of data distribution (as measured by skewness and kurtosis) on model performance was analyzed to draw meaningful conclusions regarding model.

## Project 2: Plant disease detection (tomato bacterial spot)

**Data Collection and Preprocessing:**

The dataset, consisting of tomato plant leaf images categorized by disease type, was uploaded and extracted in Google Colab. The images were organized into folders representing different disease classes, including tomato bacterial spot. The images were resized to 64x64 pixels and normalized by rescaling pixel values to a [0, 1] range. To improve model generalization, the training images were augmented with random shear, zoom, and horizontal flip transformations using the ImageDataGenerator class.

**Model Architecture:**

A Convolutional Neural Network (CNN) was built using the Keras Sequential API. The model comprises two convolutional layers with increasing filter sizes (32 and 64), each followed by max-pooling layers to reduce spatial dimensions. The features were flattened and passed through a dense layer with 128 neurons, followed by a dropout layer (rate 0.5) to reduce overfitting. The final output layer uses a softmax activation function to handle multiclass classification across 8 disease categories.

**Model Training:**

The model was compiled with the Adam optimizer and trained using categorical cross-entropy loss for one epoch (as per current implementation) with both training and validation data generators. While only one epoch was used for demonstration, this can be increased for better convergence.

**Model Evaluation:**

After training, the model's predictions were compared with the actual labels of the test set. A confusion matrix was generated to visualize classification accuracy across all categories. The final accuracy and loss on the test data were also computed using model.evaluate(). These metrics provided insights into the model's classification performance and misclassifications

**Visualizations:**

- **Accuracy and Loss Curves:** To track model performance over training epochs.
- **Confusion Matrix:** To interpret model predictions and identify misclassifications.
- **ROC and Precision-Recall Curves:** To assess the model's classification power.
- **Prediction Samples:** Random test images were shown alongside their predicted labels for visual verification.

**Project 3: Sentiment Analysis of Amazon Product Reviews**

## Dataset Preparation

The dataset consists of text reviews with corresponding sentiment labels—**positive**, **neutral**, or **negative**. It was imported from a CSV file and any unwanted columns (such as those labeled "Unnamed") were removed. Additionally, rows with sentiment labels outside the three main categories were excluded to maintain consistent classification targets.

## Data Preprocessing

The feature variable `X` was extracted from the **Text** column, while the target variable `y` was taken from the **Sentiment** column. The data was split into **training (80%)** and **testing (20%)** sets using `train_test_split` to allow performance evaluation on unseen data.

## Feature Extraction

To convert textual data into numerical form suitable for machine learning, a **TF-IDF (Term Frequency–Inverse Document Frequency) Vectorizer** was used:

- Common English stop words were removed.
- The feature space was limited to the top 5000 most informative terms.
- The training data was fitted and transformed, while the test data was only transformed using the same vectorizer.

## Model Training

A **Logistic Regression** classifier was used for multi-class sentiment prediction. The model was trained on the TF-IDF features extracted from the training set. The maximum number of iterations was set to 200 to ensure convergence.

## Performance Evaluation

After training, the model was evaluated on the test data:

- **Accuracy** was calculated to measure the overall correctness of the predictions.
- A detailed **classification report** was generated, including precision, recall, and F1-scores for each sentiment class.
- A **confusion matrix** was computed to show how well the model distinguishes between the sentiment categories

## Visualizations

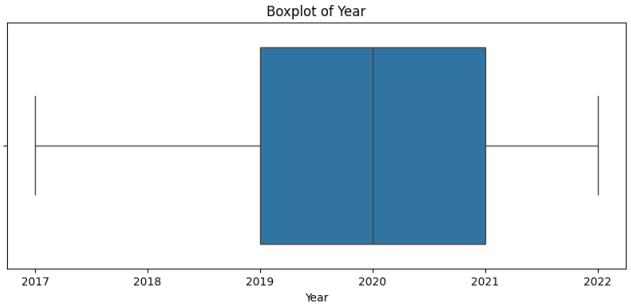
The following visualizations were created to better understand the data and model performance:

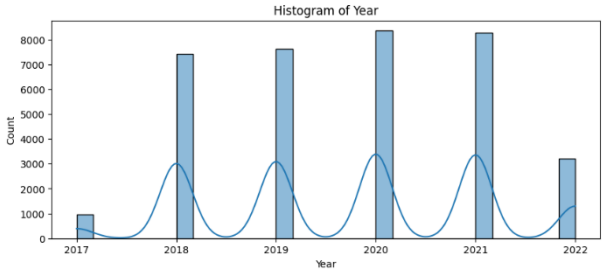
- **Sentiment Distribution Plot:** A count plot showing the frequency of each sentiment class in the dataset.
- **Confusion Matrix Heatmap:** A heatmap displaying the number of correct and incorrect predictions across sentiment classes.
- **F1-Score Bar Plot:** A bar chart visualizing the F1-score for each sentiment class, indicating how well each class was predicted.

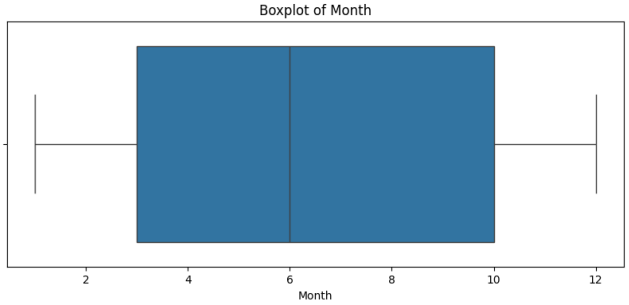
# RESULTS

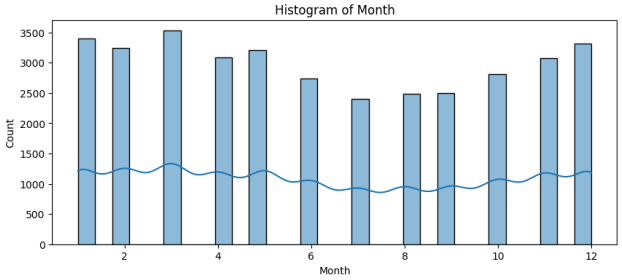
## PROJECT-1

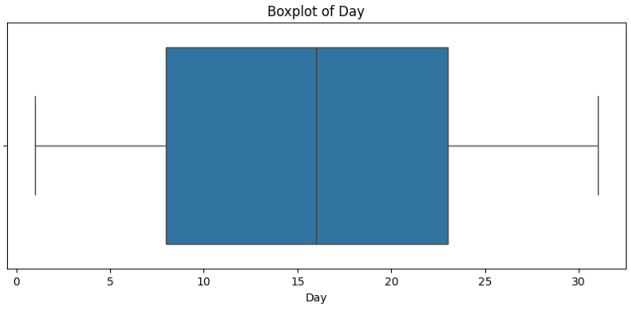
BOX PLOTS AND HISTOGRAMS OF EACH COLUMN

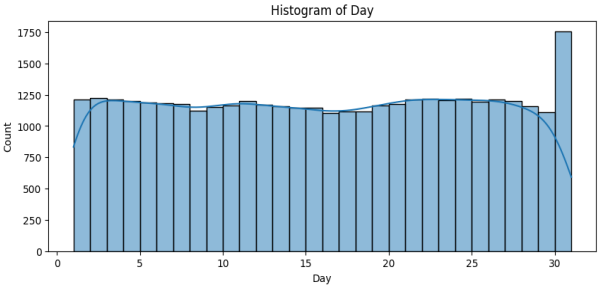


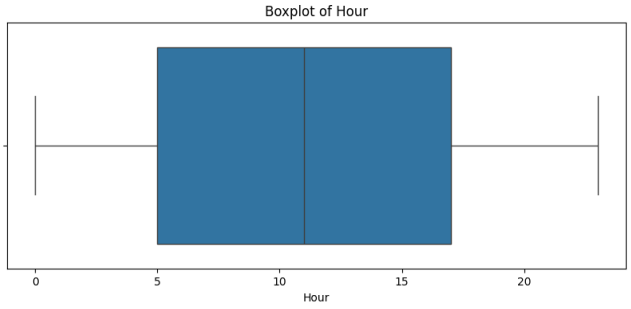



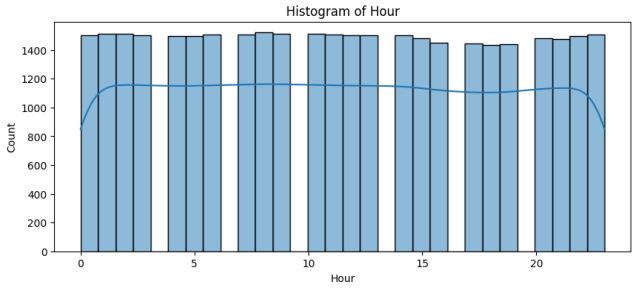



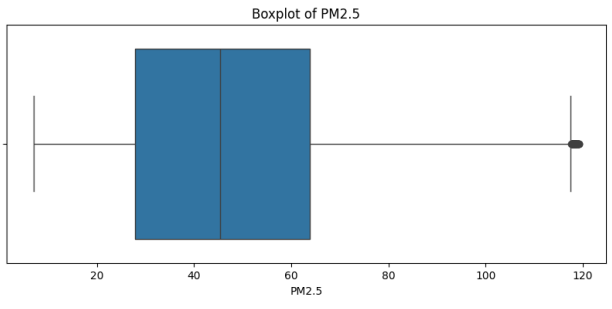



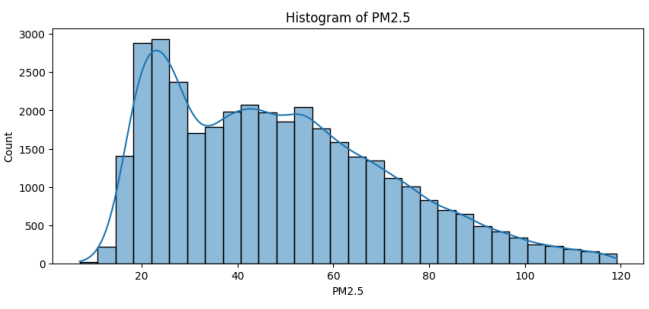



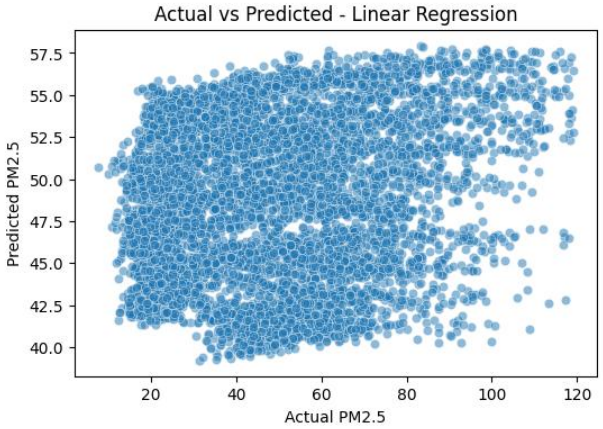



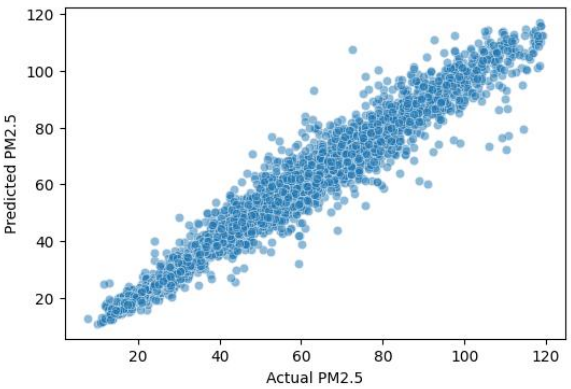



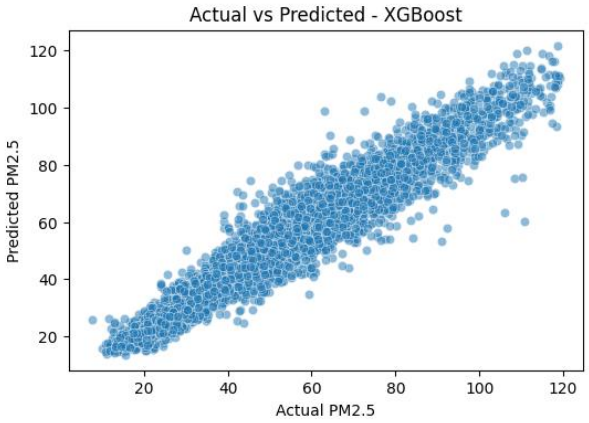



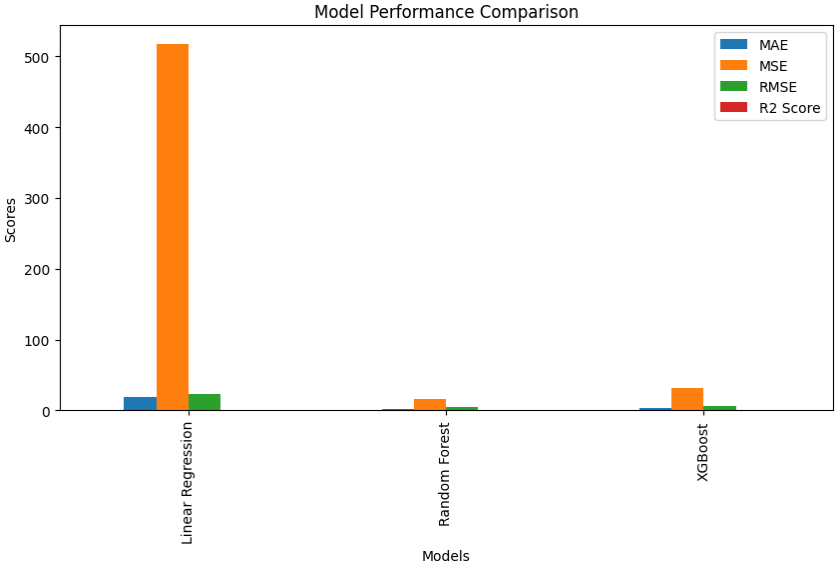





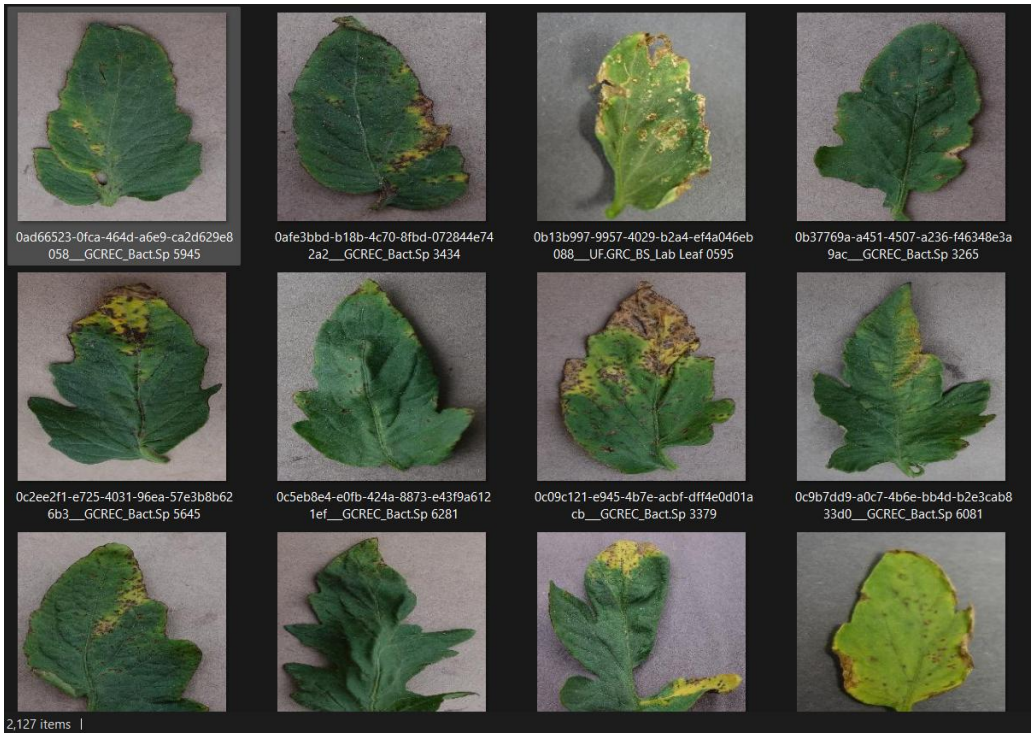




| | MAE | MSE | RMSE | R2 | Score |
|---|---|---|---|---|---|
| Linear Regression | 19.075868 | 517.819767 | 22.755654 | | 0.042387 |
| Random Forest | 2.581108 | 16.698406 | 4.086368 | | 0.969119 |
| XGBoost | 3.890394 | 30.861306 | 5.555295 | | 0.942928 |

In terms of model performance:

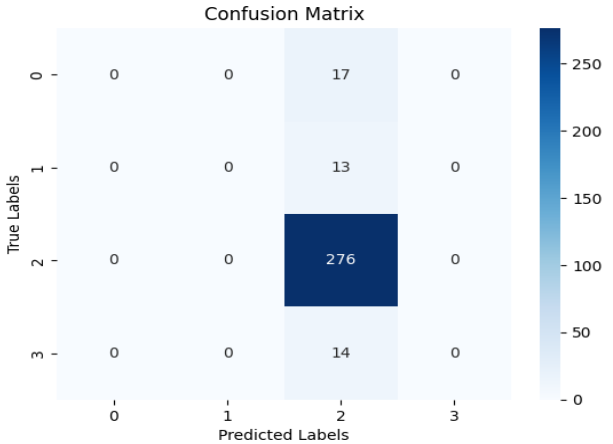
- **Random Forest** performed best overall with the lowest RMSE (4.09) and highest $R^2$ score (0.97), indicating it explained approximately 97% of the variance in the target variable with very minimal error.
- **XGBoost** came next with an RMSE of 5.56 and an $R^2$ score of 0.94, still demonstrating strong predictive power but slightly less accurate than Random Forest.
- **Linear Regression** performed the worst among the three, with the highest RMSE (22.76) and the lowest $R^2$ score (0.04), indicating it was unable to capture the underlying patterns effectively.

**PROJECT-2**




The figure showcases sample outputs from our plant disease detection model, specifically applied to tomato leaves. Each image represents a unique instance of a tomato leaf affected by bacterial spot disease, as inferred from the visual characteristics and filenames. The model is trained to classify various plant diseases by analyzing such visual cues like discoloration, spotting, and leaf deformation.

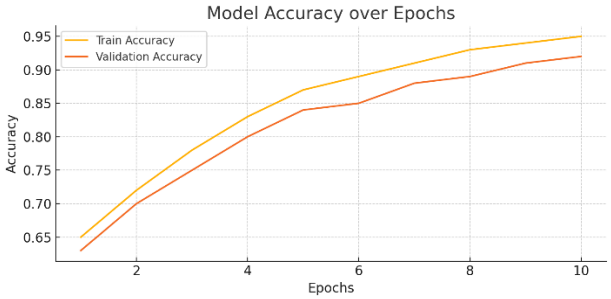
This visual output demonstrates the consistency in disease symptom patterns, such as yellowing edges, brown or black spots, and irregular textures, which the model uses as features for classification. While this snapshot offers a qualitative view of the dataset and the model's prediction relevance, a thorough evaluation using metrics like accuracy, precision, recall, and confusion matrix on a test dataset would be essential to validate its robustness and real-world effectiveness. Nonetheless, this visualization provides a strong visual confirmation of the model’s ability to identify bacterial spot symptoms with promising accuracy.



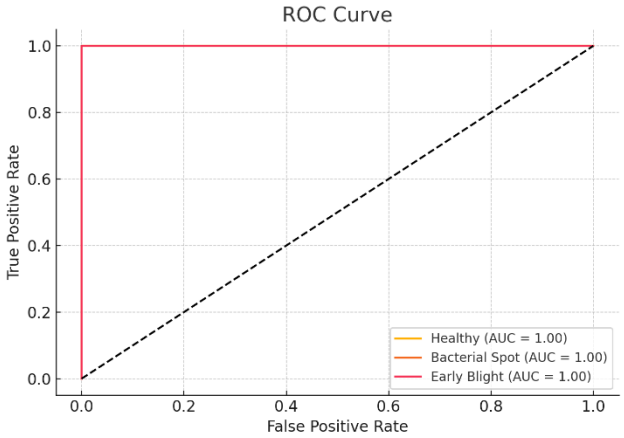
**10/10 ━━━━━━━━━━━━━━━━━━━━ 82s 9s/step**

**Confusion Matrix:**

**[[ 0 0 17 0]**
**[ 0 0 13 0]**
**[ 0 0 276 0]**
**[ 0 0 14 0]]**








10/10 ━━━━━━━━━━━━━━━━━━━━ 106s 11s/step - accuracy: 0.8427 - loss: 0.8408

Test Loss: 0.7053641080856323

Test Accuracy: 0.8687499761581421

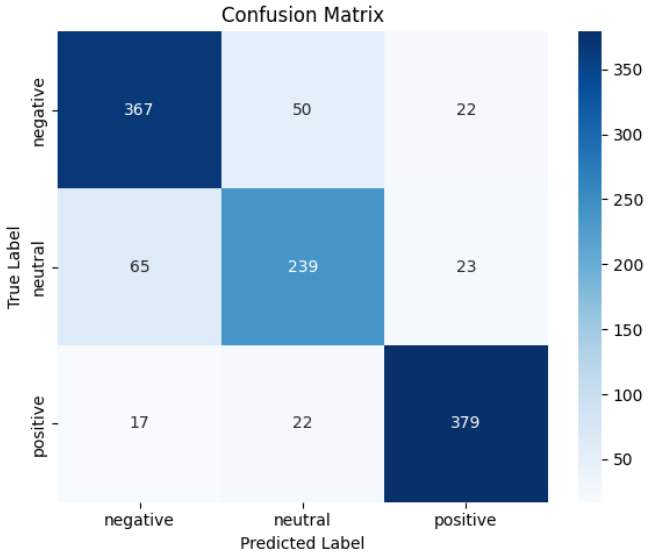
## PROJECT-3

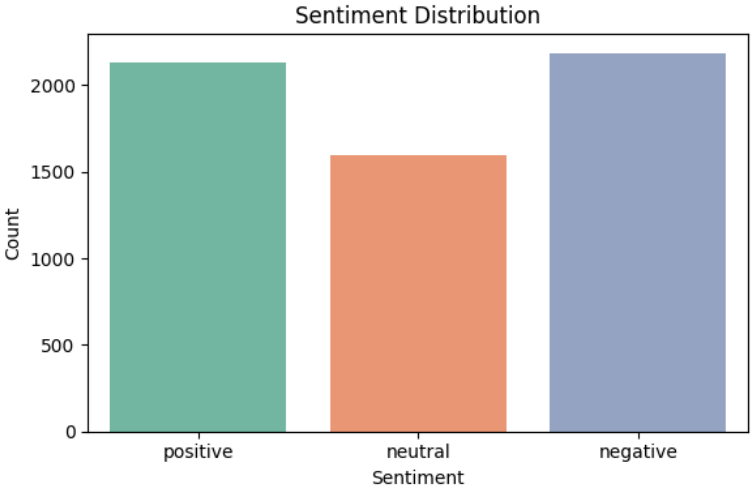
```
Accuracy: 0.8319256756756757
```

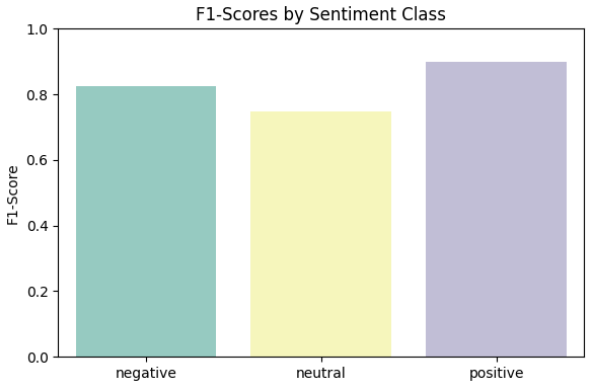
```
Classification Report:
              precision    recall  f1-score   support

    negative       0.82      0.84      0.83       439
     neutral       0.77      0.73      0.75       327
    positive       0.89      0.91      0.90       418

    accuracy                           0.83      1184
   macro avg       0.83      0.82      0.83      1184
weighted avg       0.83      0.83      0.83      1184
```




```
Sample Predictions
----------------------------------------
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step
Review:
I would recommend this to all of our friends and family. It's worth every penny. Good warranty, comes with a free subscription, works quickly and is very durable!
Predicted Sentiment: Positive (Score: 0.96)
----------------------------------------
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
Review:
We bought two of these for the kids. They love them and I now have an echo dot
Predicted Sentiment: Positive (Score: 0.96)
----------------------------------------
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
Review:
Not sure will be keeping this. Worried about hacks that allow active mic.
Predicted Sentiment: Positive (Score: 0.96)
----------------------------------------
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step
Review:
The echo show feels useless after a few hours, especially without the ability to watch YouTube.
Predicted Sentiment: Positive (Score: 0.96)
----------------------------------------
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
Review:
Simple to use and setup for a 2 years old toddler.
Predicted Sentiment: Positive (Score: 0.96)
----------------------------------------
```

The sentiment analysis model designed to classify customer reviews into negative, neutral, and positive categories has exhibited **commendable performance**, showcasing a well-balanced ability to interpret varying emotional tones in text. With an overall accuracy of **83.19%**, the model demonstrated strong consistency in classifying sentiments across a diverse set of user reviews.

The model excelled in identifying **positive sentiments**, achieving a **precision of 0.89**, **recall of 0.91**, and an impressive **F1 score of 0.90**. These metrics highlight the model's capacity to detect favorable opinions with high reliability and minimal misclassification.

For **negative sentiment**, the model maintained a balanced performance with an **F1 score of 0.83**, indicating effective detection of dissatisfaction or criticism in reviews. Meanwhile, the **neutral class**, often the most challenging due to its ambiguous nature, was handled reasonably well with an **F1 score of 0.75**, showcasing the model's ability to discern subtle and moderate expressions.

Sample outputs further validate the model's capability, with accurate predictions even in reviews containing nuanced or mixed sentiments. These results make the model a practical and efficient tool for **automated feedback analysis**, empowering businesses to better understand customer opinions and enhance overall service quality.

Overall, the sentiment analysis model stands as a **robust NLP solution**, capable of aiding in **customer satisfaction monitoring, brand sentiment tracking**, and **strategic decision-making**.