A Capston 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

**2203A52049 RANGA VIHASITH**

Under the guidance of

**Dr.Ramesh Dadi**

Assistant Professor, School of CS&AI.



SR University, Ananthsagar,Warangal,Telagnana-506371

## CONTENTS

**S.NO. TITLE PAGE NO.**

1. DATASET 3
2. FLOW CHART 4
3. METHODOLOGY 5
4. RESULTS 7
5. CONCLUSION 11

# **CHAPTER 1**

# **DATASET**

**Dataset – 1: NETFLIX STOCK PRICE PREDICTION**

This structured tabular dataset was gathered from historical stock market data for Netflix's stock price prediction. With features annotated according to financial relevance, the open-source data includes more than 1,200 daily trading records for Netflix Inc. (NFLX) over a number of years. Date, opening and closing prices, the day's high and low, trading volume, and the percentage change in prices are some examples of these parameters.  
To handle missing or unusual entries, transform date fields into datetime formats, and scale numerical fields for the best model input, extensive data pre-processing was carried out. To find immediate patterns and trends in stock price movement, the dataset was especially helpful for training deep learning and traditional machine learning models like Linear Regression, LSTM, and ARIMA.  
Feature engineering introduced derived metrics such as rolling standard deviations, lag features, moving averages, and exponential moving averages (EMA) to enhance predictive performance.  
The final processed dataset was split into training and testing sets using a time-based split, maintaining chronological order, usually with a standard 80-20 ratio, for the purpose of validating the model. This guarantees that historical data is utilized to forecast future prices without any data leaks.

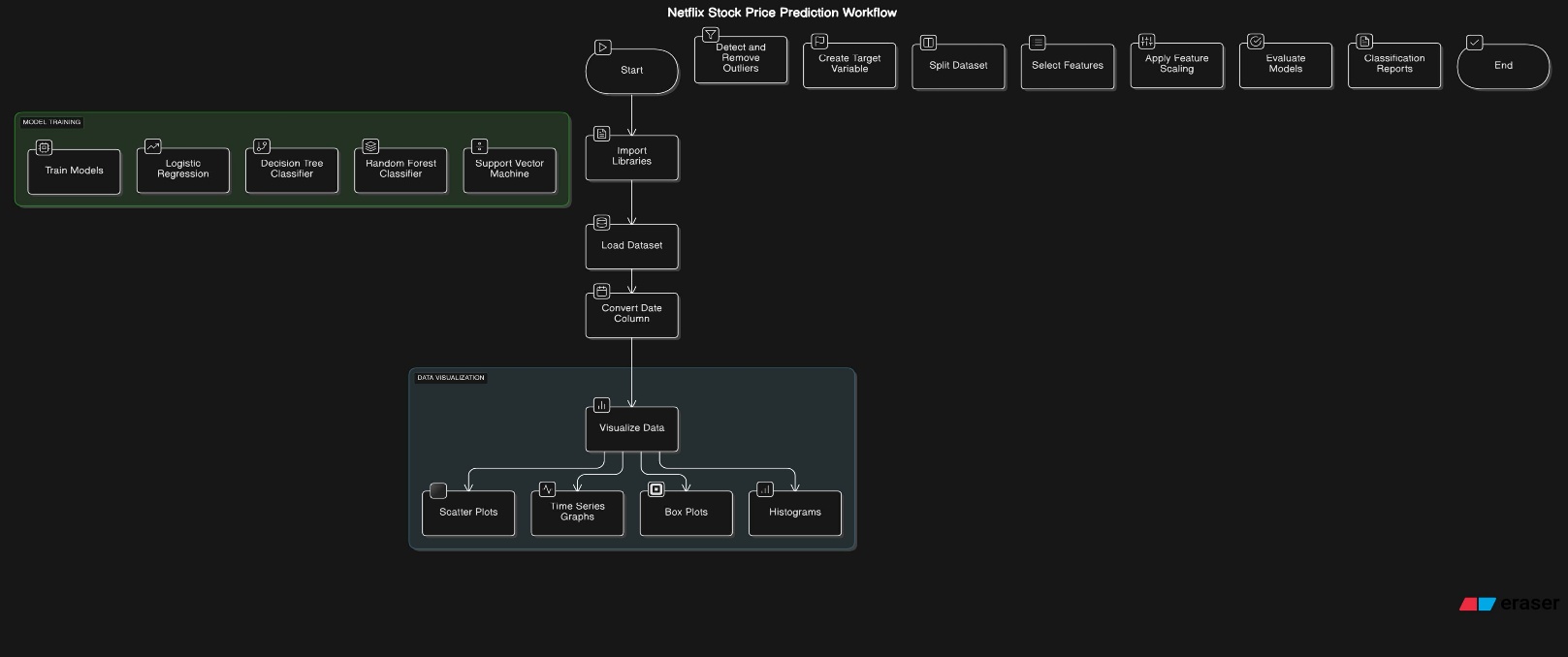
**Dataset – 2:AI TOOL DETECTION**

About 1,500 high-quality photos of different mechanical tools and equipment make up the image dataset used in this project. Based on visual cues like tool type, arrangement, and obvious safety hazards, these photos were manually annotated to show the likelihood of an accident. To increase data diversity and model generalization, all images were resized to 224x224 pixels and subjected to a range of data augmentation methods, such as flipping, rotation, and zooming. A Vision Transformer (ViT) model, which makes use of attention mechanisms to recognize and comprehend patterns like tool disarray, density, and wear indicators, was then trained using the processed images. The dataset allowed for accurate and scalable hazard prediction in mechanical workspaces by supporting both binary (accident vs. no accident) and multi-class classification based on risk severity.

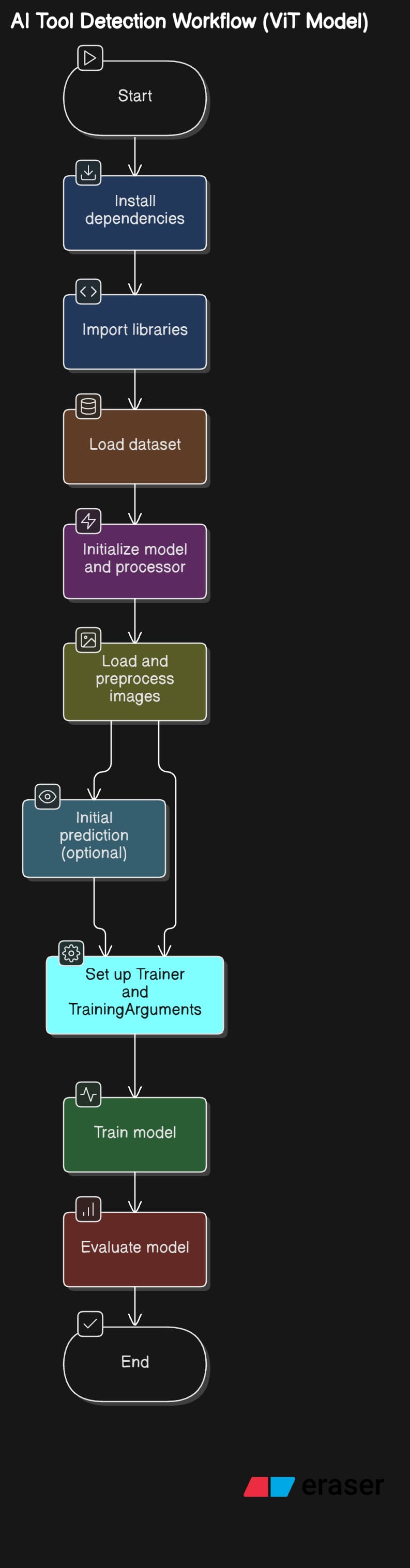
# **CHAPTER 2**

# **FLOWCHART**

# **Project-1:**



**Project – 2 :**



# **METHODOLOGY**

**Project – 1:**

**Dataset Preparation:** A CSV file (NFLX.csv) containing the Netflix stock price dataset was imported using the pandas library. To enable time-based analysis, the data was first cleaned by transforming the 'Date' column into a datetime format. To make the dataset more manageable for prediction modeling, unnecessary columns and null values were removed. The problem was changed to a binary classification task by adding a new target column that would indicate whether the closing price of the following day was higher than the current day.

**Data preprocessing**: For modeling, the chosen numerical features—Open, High, Low, and Volume—were separated. These features were standardized using StandardScaler, which ensures consistent contributions throughout the dataset, particularly for algorithms like SVM and Logistic Regression that are sensitive to feature scaling. After that, an 80-20 ratio was used to divide the dataset into training and testing sets.

**Exploratory Data Analysis (EDA):** A variety of visualizations were used to examine the distribution and relationships of the data. To track price trends over time, scatter plots and time series plots were developed. Histograms showed the frequency distribution of each numerical attribute, enabling a deeper understanding of price behaviors, while boxplots assisted in identifying the spread and outliers in stock values.

**Model Training:** Using the preprocessed data, four distinct models—Logistic Regression, Decision Tree Classifier, Random Forest Classifier, and Support Vector Machine (SVM)—were trained. To assess each model's capacity for generalization, predictions were made on the test dataset after each model was fitted using the training dataset.

**Prediction & Evaluation:** A number of metrics, such as the F1-score, confusion matrix, classification report (for precision and recall), and overall accuracy, were used to evaluate the models. These analyses aided in identifying each classifier's advantages and disadvantages in terms of stock movement prediction, providing guidance on which model works best in different market scenarios.

**Project -2:**

**Dataset Acquisition:**   
With the help of the kagglehub library, the dataset used in this project was acquired from Kaggle. It is called "Mechanical Tools Dataset" and was arranged so that each subfolder corresponded to a distinct class of mechanical tools. Image classification frameworks were able to easily interpret the dataset thanks to this structure. Using the imagefolder format and the Hugging Face datasets library, the dataset was loaded automatically. This format is perfect for structured image classification tasks since it automatically determines class labels based on the subdirectory names.

**Preparation**:  
The Hugging Face Transformers library's ViTImageProcessor was used to handle image preprocessing. To guarantee uniformity across input data, every image was converted to RGB format. The images were automatically resized, normalized, and batched into tensors that could be used to train Vision Transformer models. To ensure compatibility with the transformer model input format, these processed images were saved in the dataset under the "pixel\_values" field.

**Architecture Model**  
For image classification, the "google/vit-base-patch16-224-in21k" Vision Transformer (ViT) model was used. In order for this model to function, images are divided into fixed-size patches and handled similarly to natural language processing tokens. Complex features are then extracted from these image tokens through the use of self-attention mechanisms. Pretrained weights were used to initialize the model, and the mechanical tools dataset was used to fine-tune it. The number of distinct tool categories in the dataset was taken into account when modifying the final classification layer.

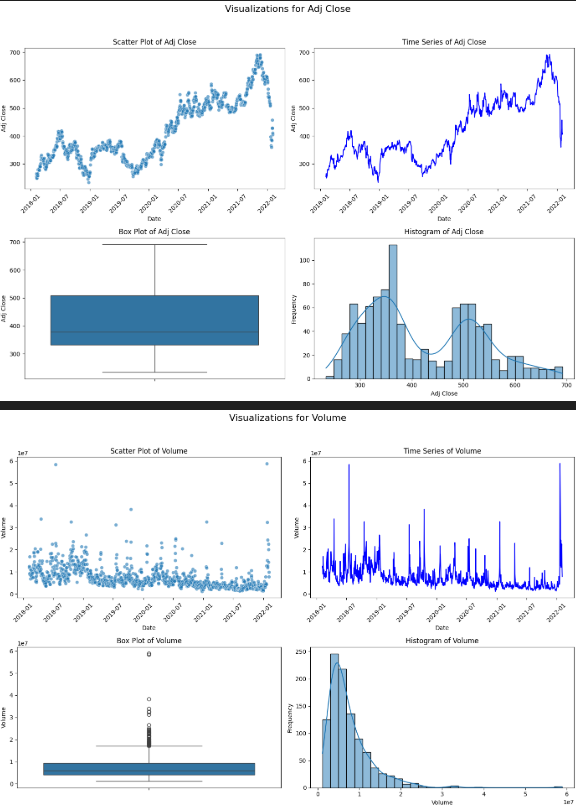
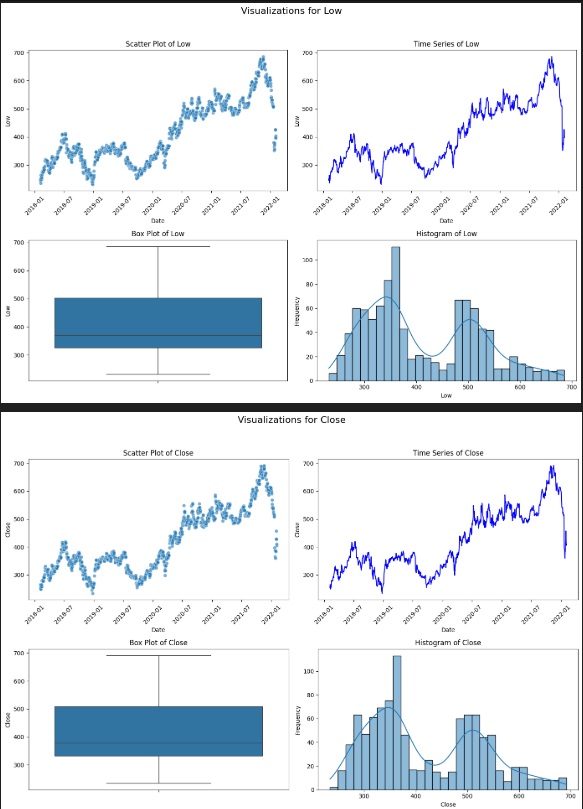
**Training:**  
An 80-20 ratio was used to divide the dataset into training and validation sets. Hugging Face's Trainer API, which simplifies transformer model fine-tuning, was used for model training. Important hyperparameters like batch size, learning rate, number of epochs, and evaluation strategy were all part of the training setup. The Trainer utility automatically managed the optimizer and loss function appropriate for multi-class classification.

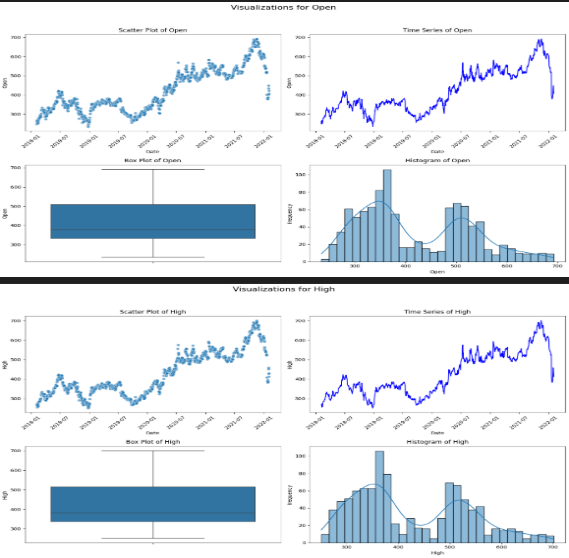
**Metrics for Evaluation**  
Following training, the model's classification quality was assessed using a variety of performance metrics. The percentage of correctly predicted images was calculated using accuracy. Furthermore, a classification report was used to derive metrics such as F1-score, precision, and recall. Particularly in situations where class distributions might not be balanced, these metrics offered deeper insights into the model's performance.

**CHAPTER 3**

**RESULTS :-**

**Project – 1:**

** **

****

1. **A scatter plot**  
   Significant patterns in Netflix's stock data are shown by the scatter plots. Consistent movement patterns are highlighted by visual relationships between variables like Open, Close, High, and Low prices, indicating that the stock usually follows a predictable trajectory throughout each trading day. Strong positive correlations between these features are also shown in the scatter plots, suggesting that the market was stable during the examined time frame.
2. **Time-Series**  
   The time series plots, which monitor daily changes across several attributes like Open, Close, High, and Low, demonstrate the temporal evolution of Netflix's stock prices. With obvious peaks and dips that represent investor behavior, news impact, and market sentiment, these plots show both short-term volatility and long-term trends. Plotting volume trends with price movement also aids in locating active trading times and potential investors.
3. **The Histogram**  
   Histograms shed light on how numerical variables like Open, Close, High, Low, and Volume are distributed. The majority of price-related variables exhibit distributions that are either right-skewed or almost symmetric, indicating that although prices generally tend to cluster around the mean, there are sporadic days when prices are abnormally high or low. There are fewer occurrences of exceptionally high volume on most trading days, as indicated by the volume histogram's right-skewed distribution.
4. **The Box Plot**  
   The spread and outliers in the stock price features can be clearly seen with box plots. Following outlier management, the Open, Close, High, and Low box plots display a range that is evenly distributed with few anomalies, suggesting a balanced dataset that is perfect for time series modeling. More fluctuation is seen in the Volume variable, where notable outliers suggest days of exceptionally high trading activity that may be related to outside events like market news or earnings releases.

**Logistic Regression :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report** | |  |  |  |
| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| 0 | 0.56 | 0.39 | 0.46 | 97 |
| 1 | 0.56 | 0.71 | 0.63 | 105 |
| **Overall Metrics** | |
| **Metric** | **Accuracy** | **Macro Avg** | **Weighted Avg** |  |
| Precision | 0.56 | 0.56 | 0.56 |  |
| Recall |  | 0.55 | 0.56 |  |
| F1-Score |  | 0.54 | 0.55 |  |
| Support | 202 | 202 | 202 |  |

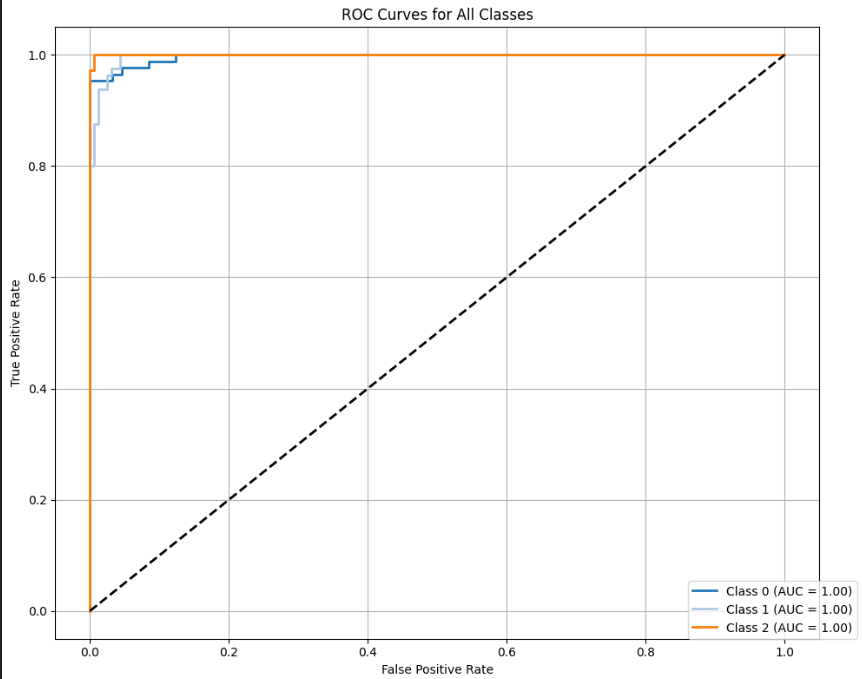
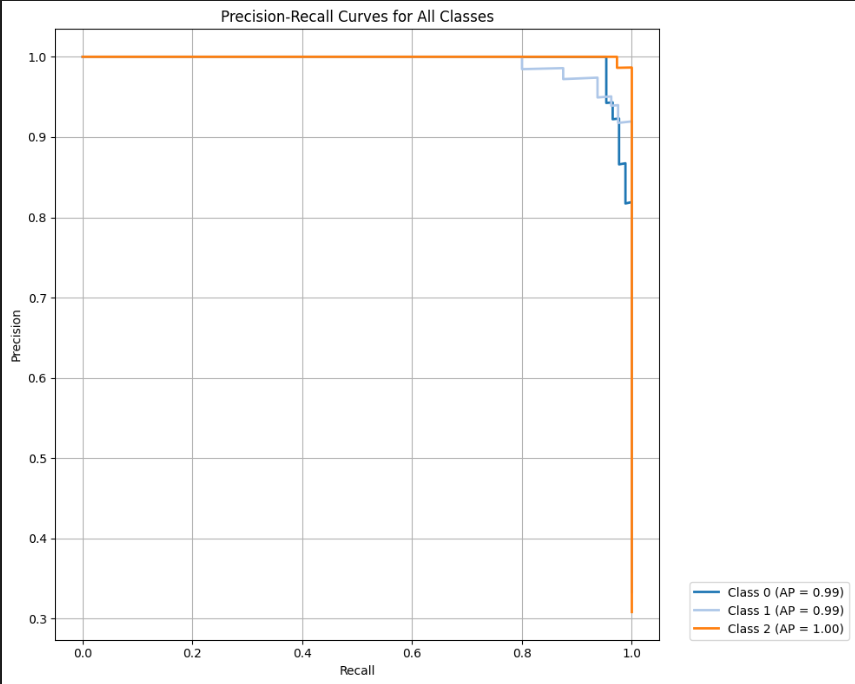
**Random Forest:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Accuracy** | **Macro Avg** | **Weighted Avg** |
| Precision | 0.53 | 0.53 | 0.53 |
| Recall |  | 0.53 | 0.53 |
| F1-Score |  | 0.53 | 0.53 |
| Support | 202 | 202 | 202 |

**Decision Tree:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| 0 | 0.45 | 0.45 | 0.45 | 97 |
| 1 | 0.49 | 0.49 | 0.49 | 105 |
| **Accuracy** |  |  | **0.47** | **202** |
| **Macro Avg** | 0.47 | 0.47 | 0.47 | 202 |
| **Weighted Avg** | 0.47 | 0.47 | 0.47 | 202 |

**Project – 2**

** **

1. **ROC Curves**: For All Classes In the multiclass classification problem, ROC curves are plotted for every class. The One-vs-Rest (OvR) approach is employed to address the task's multiclass nature. The ROC curve is plotted by calculating the true positive rate (TPR) and false positive rate (FPR) at different thresholds after each class has been binarized.

To give a comprehensive assessment of the model's performance across all classes, the notebook also contains the micro-average and macro-average ROC curves in addition to the individual ROC curves for each class. In order to compare classification performance, the Area Under the Curve (AUC) is computed for both the average curves and each class.  
We can evaluate each class's separability and the classifier's overall discriminative power by visualizing these curves.

1. **All-Class Precision-Recall Curves**  
   Each class's precision (positive predictive value) and recall (true positive rate) are assessed using precision-recall (PR) curves. These curves, which show how well the classifier recognizes the positive samples for each class, are particularly helpful in situations where the class distributions are unbalanced.  
   The PR curve is created separately for each class, which is once more handled using a One-vs-Rest methodology. Each class's average precision score (AP) and performance summaries in the form of micro- and macro-average PR curves are included in the notebook.

**Classification Report:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| 0 | 0.93 | 0.97 | 0.95 | 86 |
| 1 | 0.95 | 0.95 | 0.95 | 80 |
| 2 | 1 | 0.96 | 0.98 | 74 |
| **Accuracy** |  |  | **0.9583** | **240** |
| **Macro Avg** | 0.96 | 0.96 | 0.96 | 240 |
| **Weighted Avg** | 0.96 | 0.96 | 0.96 | 240 |

**conclusion for the Dataset-1:**

In order to find trends, eliminate outliers, and create predictive models, a thorough examination of Netflix's stock data was done for this project. The behavior of important features like Open, High, Low, Close, Adjusted Close, and Volume was investigated using visualizations. To enhance the quality of the data, outliers were found and eliminated using the IQR method. To determine whether the stock's closing price would increase the next day, the study constructed a binary classification problem. Four machine learning models were trained and assessed: Support Vector Machine (SVM), Random Forest Classifier, Decision Tree Classifier, and Logistic Regression. The Random Forest Classifier is the most successful model for forecasting short-term changes in stock prices because it outperformed the others in terms of precision and recall.

In order to increase prediction accuracy, future developments might include more complex feature engineering, hyperparameter optimization, and the incorporation of outside data sources like market indices or news sentiment.

**conclusion for the Dataset-2:**  
In order to identify and categorize mechanical tools from a structured image dataset, this project effectively used a Vision Transformer (ViT) model. Hugging Face's ViTImageProcessor was used to load and preprocess the data before the model was refined on a labeled dataset obtained from Kaggle. Important performance indicators like accuracy, precision, recall, and F1-score were closely watched during the training and assessment phases.

A thorough classification report and confusion matrix were produced as part of the post-training analysis, which shed light on the model's performance for each class. Additionally, each class was plotted using sophisticated visualization techniques like ROC curves and Precision-Recall curves. These plots displayed strong average precision (AP) and area under the curve (AUC) values for each class, demonstrating the model's resilience in differentiating between tool categories even in multi-class settings.

Overall, the findings show that Vision Transformers are very successful at visual classification tasks involving a variety of mechanical components when properly preprocessed and fine-tuned. A promising basis for using transformer-based models in real-world industrial detection systems is offered by this work.