Summer Internship Project Report
on
WASTE SEGREGATION
At

SCIENTIFIC ANALYSIS GROUP (SAG)
Defence Research and Development Organisation

SUBMITTED BY-

KAKUL GUPTA (00602212022)
PRAGYA ARORA (01202212022)
RAMAN SHARMA (01502212022)
NOOR YESHFEEN (02302212022)

Under the supervision of

Mrs. Pooja Yadav

In Partial Fulfilment of the Requirements for the Degree of

Masters of Technology (Artificial Intelligence and Data Science)

Indira Gandhi Delhi Technical University for Women

ACKNOWLEDGEMENT

I avail this opportunity to express my profound sense of sincere and deep gratitude to those who have played an indispensable role in accomplishing the project work given to me by providing their willing guidance and help.

Firstly, I express my sincere gratitude to **Ms. Anu Khosla, Director, Scientific Analysis Group**, for allowing me to carry out the project work in this prestigious research and development center and gain valuable experience

I am blissful to express my deep sense of gratitude to **Mrs. Pooja Yadav** for discussing the module of the project systematically. His constant interaction, expert guidance, and valuable suggestions help me to complete this project successfully.

I also wish to thank **Mr. Arun Chaudhary** for his kind and continual support and constructive suggestions given during the course of this project.

Last, but not least, I would also like to thank all the members of the SAG for giving me easy access to all the resources like PC, etc. needed for the successful completion of the project.

1. Introduction

Solid waste management is a concerning problem for people all across the world. In our country, tons of waste are generated and it is difficult to segregate waste between biodegradable and non-biodegradable waste. Harmful chemicals have become a constant companion of modern life. They are used to sanitize houses, power bulbs, and tube lights and even find refuge in medicines, ointments, and the like. While these potentially dangerous products are handled with extreme care at home, similar caution is not exercised when throwing them out with household garbage.

Till now, in smaller towns and villages, the citizens either do not segregate their waste or do not know about it. Even in big cities, people don't have the time and patience to segregate them into biodegradable and non-biodegradable waste. Even though we have dustbins labeled for waste segregation but the waste ends up getting mixed.

We propose to build a waste segregation system using CNN where we will classify the waste into biodegradable and non-biodegradable waste. Further, we will divide this waste into various sub-categories and study our model inner technique using AI Explainability models like LIME, SHAP, and GRAD-CAM which will add novelty to this project. Thereafter, we aim to create an application that can detect the contents of the waste and help people to segregate between the different kinds of waste. This will be beneficial for recycling different materials. Further, this can also be used in conveyor belts to segregate tons of waste collected by the municipality. We can adjust scanners in conveyor belts to detect and segregate biodegradable waste inside non-biodegradable waste like poly bags or vice-versa. In the future, we can detect categories based on atom arrangement.

2. Literature Review

The work on solid waste management is being done for the past many years since it is a very sensitive subject for the safety of our mother Earth and the environment in general.
In paper [1] CNN based waste classifiers were explored which included (VGG16, ResNet-50, MobileNet V2, and DenseNet-121) in classifying waste types of 9200 municipal solid waste images. The highest classification accuracy achieved was 94.86 percent using ResNet 50 classifier.
In paper [2], hierarchical deep learning is used for waste detection and classification in food trays In Reference [3] 1D CNN method is used to find features from raw data and achieve high classification accuracy. Here the highest classification accuracy achieved was 92.40 Reference[4] is the state-of-the-art review where solid waste management and generation techniques are analyzed. Various aspects are analyzed like the characteristics of municipal solid waste, the need for energy recovery, and various techniques to produce waste. It also highlights India faces for solid waste management Reference [5] classifies waste using cloud computing into nine categories (kitchen waste, other waste, hazardous waste, plastic, glass, paper or cardboard, metal, fabric, and other recyclable waste)

3. Methods

3.1 Data Collection

We have taken our dataset from Kaggle. Source of our dataset: Waste Segregation Image Dataset,2022[6]. This Image dataset has been divided into train and validation folders and they are further divided into 8 classes.
For Biodegradable we have 4 classes: paper, leaf, food, wood
For Non-Biodegradable we have 4 classes: waste, plastic bags, plastic bottles, metal cans

3.2 Data Pre-processing

In data preprocessing, the features in the dataset will be checked before proceeding with the modeling. In data pre-processing, the following tasks were carried out:
DATA CLEANING: As a part of the data cleaning, the images with less resolution were found and removed. The method used for finding the blurred images was Laplacian variance. The images with variance less than the threshold were removed. The threshold value taken is: 5
DATA ANALYTICS AND VISUALIZATION: The images in the dataset were fixed with an image size of 512x512 pixels. Since the dataset is in image form, it can only be analyzed by converting it into a 3x3 matrix which denotes the RGB channels.

The dataset was visualized using various plots as follows:

Figure 1. shows the number of images for each kind of waste. It can be inferred that the maximum number of images belong to food waste which is around 10000. We would require to balance our dataset to improve the performance of our model.

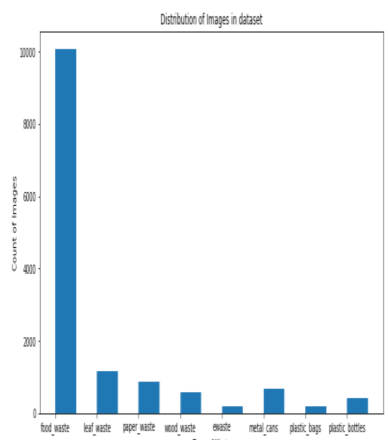


Figure 1: Distribution of images in the dataset

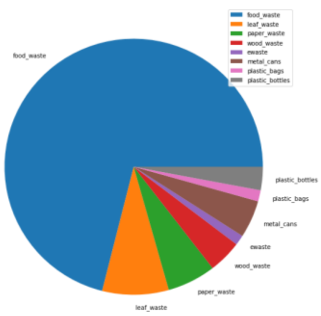


Figure 2: Pie Chart for dataset

Figure 2. is a pie chart visualization of the above bar chart. It can be inferred from the graph that 70 percent of the images belong to the food waste category.

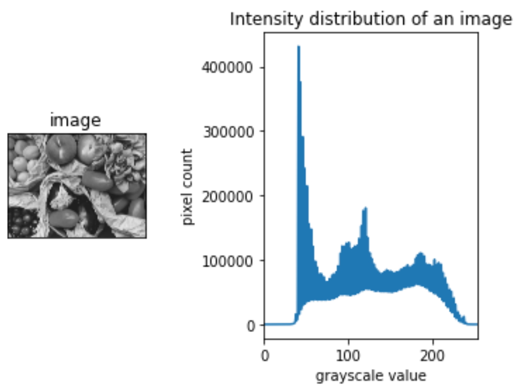


Figure 3: Intensity Distribution of grayscale image from the dataset

Depiction: By looking at this histogram [Figure 3] we can get an intuition about the brightness, contrast, and intensity distribution of an image. The maximum number of pixels in the image has a pixel value of around 45. Higher the pixel value, the lesser the intensity. So, for black color, pixel values are (0,0,0). It can help us in the classification of waste.

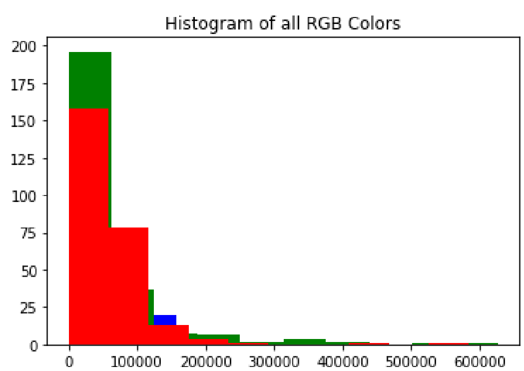


Figure 4: Intensity Distribution of a colored image

Depiction: [Figure 4] is a histogram of RGB values in an image. It tells us about the presence of RGB colors in an image. If the image has more pixels of the green channel, it might contain a leaf and hence can be predicted as leaf waste.

4. Training Phase

4.1 Train-Test-Val split

Before splitting, our dataset is divided into two sets-train set and validation set. We have used the train set to divide it into train and test sets further. 85.2 percent for the training set, 7.4 percent for the test dataset, and 7.4 percent for the validation dataset as seen in Figure 5.

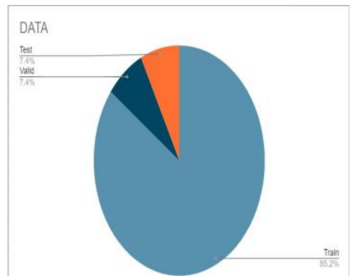


Figure 5: PERCENTAGE OF TRAIN, VALID, TEST DATA 4.2

Algorithms Used and Their Analysis

Model	**Algorithm Used**	**Statement**	**Accuracy**
Model 1	Logistic Regression	We have used the one vs all approach for the implementation of logistic regression for the multi-class data set	52 percent
Model 2	Support Vector Classifier	After dividing the multi-class classification problem into multiple binary classification problems, SVC performed well	71.6 percent
Model 3	Decision Tree	Gini criteria are used to prune the decision tree to avoid overfitting	64 percent
Model 4	Random Forest	The criteria used is Entropy and the average of the output of 100 decision trees is taken	89 percent
Model 5	CNN with SVM as final layer	The activation function for the first 3 layers was taken as'relu' and 'softmax' for the final layer. Number of epochs=20; kernel size=3	77 percent

Table 1: **Waste Classification Models and their Analysis**

5. Confusion Matrices

5.1 Decision Tree

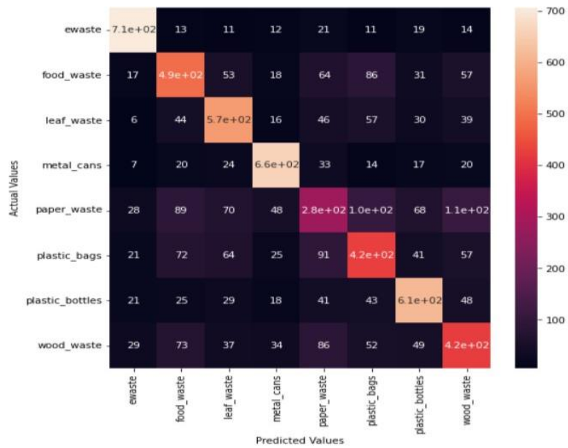


Figure 6: Confusion Matrix for Decision Tree Classifier

5.2 Random Forest

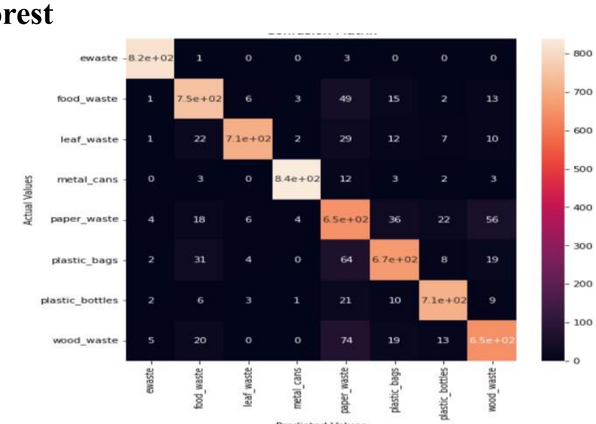


Figure 7: Confusion Matrix for Random Forest Classifier

- The diagonal elements represent the correctly predicted images by the model in Fig 6, 7.
- The cell values with zero signify that model doesn't confuse the samples of the corresponding two predicted and actual classes for that cell.
- Example- M represents the matrix, i is the row number and j is the column number. Then cell $M_{(1,3)}$ implies that the model learned the classification boundary of e-waste and leaf waste well.
- To improve model performance, we must focus on the paper waste class since the wrongly predicted values for this class are maximum. So, the misclassification rate for the paper waste class is maximum.
  Falsely predicted samples for 'paper waste'=4+18+6+4+36+22+56=146 in Fig 7.

6. ROC-AUC Curve

6.1 Random Forest ROC curve

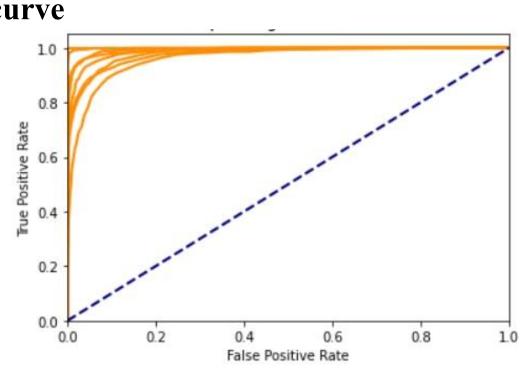


Figure 8: Receiver Operating Characteristic curves for each class

Since we can see the ROC curves for each class are shifted toward the top (i.e., higher values) of the True positive rate(y-axis) and left (i.e., lower values) of the False positive rate(x-axis), this shows the area under the curve (AUC) for each class is nearly equal to 1 or at least more than 0.5 Hence, we can say our model performed exceptionally well.
AUC $\propto$*Performance*

6.2Classification Report

	precision	recall	f1-score	support
0	0.98	1.00	0.99	821
1	0.88	0.89	0.89	837
2	0.97	0.89	0.93	789
3	0.99	0.97	0.98	862
4	0.72	0.82	0.77	795
5	0.88	0.84	0.86	795
6	0.93	0.93	0.93	766
7	0.85	0.83	0.84	778
accuracy			0.90	6443
macro avg	0.90	0.90	0.90	6443
weighted avg	0.90	0.90	0.90	6443

Figure 9: Classification Report for Random Forest Classifier

- The classification report [figure 9] contains evaluation metrics for 0-7 classes.
- In our classification problem, we want both precision and recall to be high. Hence, we will take the f1-score into account,
- *f1-score = harmonic mean of precision & recall*

7. Data Augmentation

Data Augmentation is used while training the data to avoid overfitting by adding a sample of images that are flipped, rotated, and zoomed. Our model will randomly flip the images while training the data. It will randomly rotate the images in the 20 percent range and will randomly zoom the images with a height and width factor of 30 percent and 20 percent respectively.

8. Class Balancing

Our training dataset was not balanced, so before applying models, we balanced our dataset using the oversampling technique - SMOTE (Synthetic Minority Oversampling Technique).
SMOTE is an oversampling technique where synthetic samples are generated for the minority class. This algorithm helps to overcome the overfitting problem posed by random oversampling. It focuses on the feature space to generate new instances with the help of interpolation between the positive instances that lie together.

9 Ensemble Learning

To increase the accuracy of our model, we have trained our dataset using two methods of ensemble learning:

1. Random Forest
2. Extra Tree Classifier
3.

9.1 Random Forest

Random Forest uses bagging to select different variations of the training data to ensure decision trees are sufficiently different. In our model, we have used 100 decision trees. After applying random forest, we got an accuracy of 89 percent.

9.2 Extra Trees Classifier

A highly Randomized Trees Classifier (Extra Trees Classifier) is an ensemble learning technique that aggregates the results of multiple decorrelated decision trees collected in a “forest” to output its classification result. Extra Trees uses the entire dataset to train decision trees After applying the extra trees classifier we have got an accuracy of 91 percent.

	precision	recall	f1-score	support
0	0.97	1.00	0.98	833
1	0.87	0.87	0.87	784
2	0.96	0.89	0.92	866
3	0.99	0.98	0.99	780
4	0.72	0.81	0.77	787
5	0.83	0.84	0.83	789
6	0.94	0.92	0.93	800
7	0.88	0.85	0.87	804
accuracy			0.89	6443
macro avg	0.90	0.89	0.89	6443
weighted avg	0.90	0.89	0.89	6443

Figure 10: CLASSIFICATION REPORT

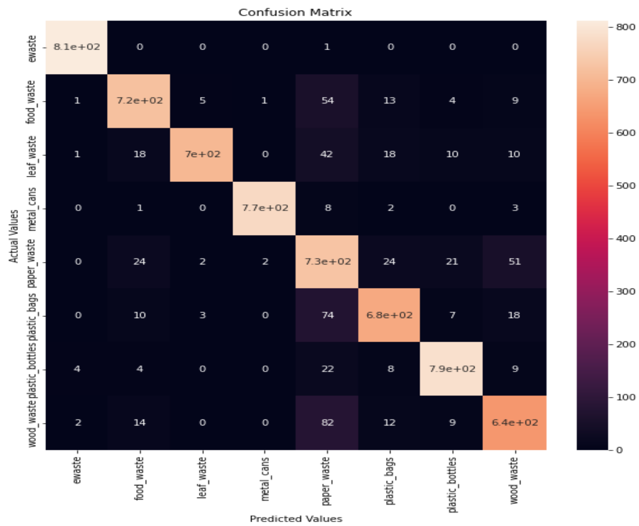


Figure 11: CONFUSION MATRIX

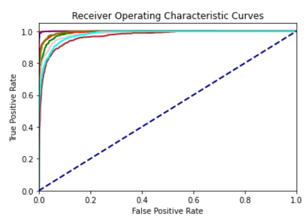


Figure 12: RECIEVER OPERATING CHARACTERISTICS

AUC for e-waste: 0.9999569604932285
AUC for food waste: 0.9905136836937535
AUC for leaf waste: 0.9949667444230692
AUC for metal cans: 0.9997199163351368
AUC for paper waste: 0.9702829749018165
AUC for plastic bags: 0.9845818229036554
AUC for plastic bottles: 0.9950696028460666 AUC for wood waste: 0.9812659764737298 One-vs-Rest ROC AUC scores:
0.989545 (macro),
0.989482 (weighted by prevalence)

10. AI EXPLAINABILITY

AI explainability refers to the ability to understand and interpret the decision-making process of an AI system. It involves providing transparent and interpretable explanations for the predictions or actions taken by AI models. Explainability is crucial for building trust, ensuring fairness, and addressing ethical concerns associated with AI systems. Here are some types of AI explainability:

a) Rule-Based Explanations:

- In rule-based explanations, AI models provide explanations in the form of logical rules or decision trees.
- These rules explicitly outline the conditions and factors influencing the model's predictions.
- Rule-based explanations offer interpretability and transparency, making it easier for humans to understand the model's decision process.

b) Feature Importance:

- Feature importance techniques help identify the input features that contribute most significantly to the model's predictions.
- They provide a ranking or score for each feature, indicating its relevance.
- Feature importance methods include techniques like permutation importance, SHAP (Shapley Additive Explanations), and LIME (Local Interpretable Model-Agnostic Explanations).

c) Model-Agnostic Approaches:

Model-agnostic approaches aim to explain any AI model, regardless of the underlying algorithm.
They treat the AI model as a black box and generate explanations by observing its input-output behavior.

- Techniques such as LIME, SHAP, and Anchors fall under this category and are commonly used to explain complex models like neural networks.

d) Natural Language Explanations:

- Natural language explanations involve generating human-readable explanations in the form of text.
- These explanations aim to provide understandable reasoning for the model's predictions.
- Natural language generation techniques, such as text summarization or generation based on model activations, can be used to generate these explanations.

e) Counterfactual Explanations:

- Counterfactual explanations provide insights into how a model's decision might have changed if specific inputs differed.
- They involve generating hypothetical scenarios or examples to illustrate how the model's predictions are affected by variations in the input features.
- Counterfactual explanations can help users understand the model's sensitivity to different inputs and potential biases.

f) Visual Explanations:

- Visual explanations involve using visualizations or graphical representations to explain AI model predictions
- Techniques like saliency maps, activation heatmaps, or attention mechanisms can highlight the regions or features in an image that influenced the model's decision.
- Visual explanations provide intuitive and interpretable insights, especially for computer vision tasks.

It's important to note that the choice of explainability technique depends on the specific AI model, application domain, and the desired level of transparency required for the system. Different techniques can be combined to provide a comprehensive understanding of the AI system's decision-making process.

In our project, we have used LIME, SHAP, and GRAD-CAM, as we need to identify which feature our model is using to predict the label.

LIME:

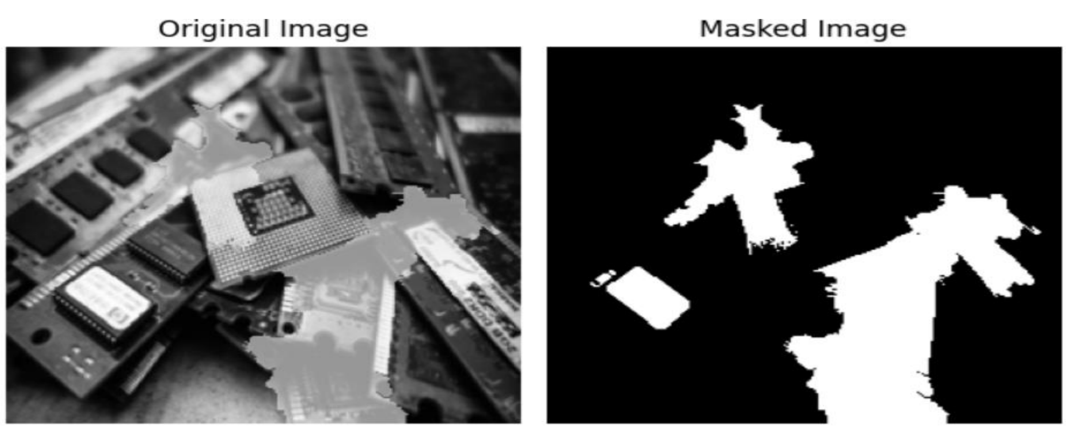
LIME stands for Local Interpretable Model-agnostic Explanations. It is a Python library based on a paper from Ribeiro et al. to help you understand the behavior of your black-box classifier model. Currently, you can use LIME for a classifier model that classifies tabular data, images, or texts.

The abbreviation of LIME itself should give you an intuition about the core idea behind it. LIME is:

Model agnostic, which means that LIME is model-independent. In other words, LIME is able to explain any black-box classifier you can think of.

Interpretable, which means that LIME provides you a solution to understand why your model behaves the way it does.

Local, which means that LIME tries to find the explanation of your black-box model by approximating the local linear behavior of your model.



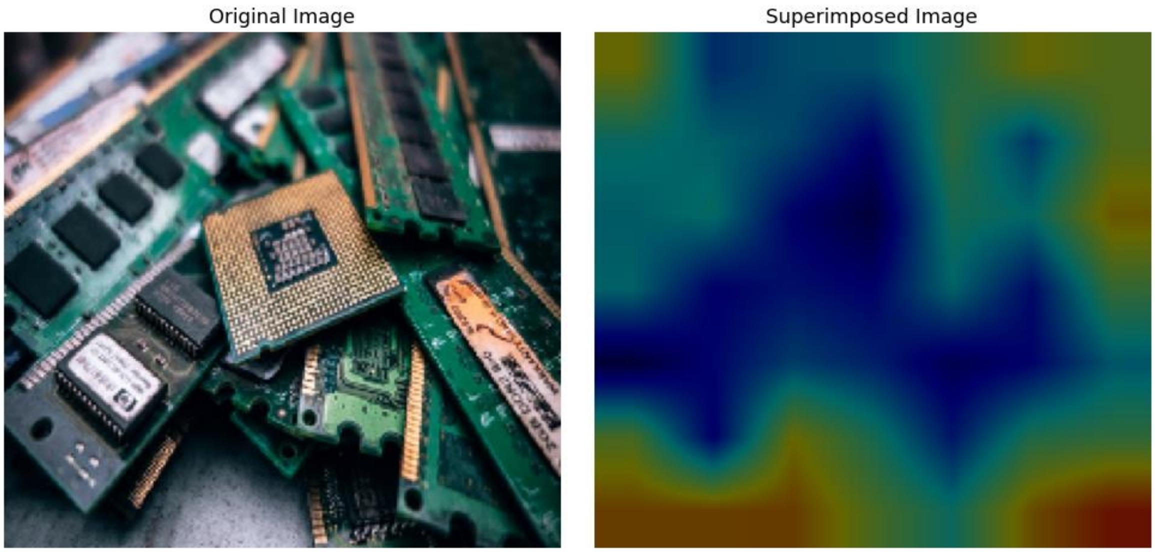
In this visualization:

Red pixels represent positive SHAP values that contributed to classifying that image as a class.

Blue pixels represent negative SHAP values that contributed to not classifying that image as that particular class.

GRAD-CAM

Grad-CAM provides a visual explanation of the model's decision-making process. It helps identify the areas the model focuses on when making predictions, offering insights into the model's behavior and enabling interpretation of its decisions.



11. Robustness

Machine learning robustness refers to the ability of a machine learning model to maintain stable and accurate performance in the face of various challenges, such as noisy or corrupted data, adversarial attacks, distribution shifts, or changes in input conditions. It involves building models that are resilient and reliable, even in the presence of uncertain or adverse conditions. Here are some aspects related to machine learning robustness:

a) Noisy or Corrupted Data:

- Machine learning models are often trained on datasets containing errors, outliers, or missing values.
- Robust models are designed to handle such noisy or corrupted data without significantly affecting their performance.
- Techniques like data cleaning, outlier detection, and imputation methods can help improve robustness in the face of noisy data.

b) Adversarial Attacks:

- Adversarial attacks involve intentionally manipulating input data to deceive or mislead machine learning models.
- Robust models are resilient against adversarial attacks and can maintain accurate predictions even when presented with examples.
- Adversarial training, where models are trained on adversarial examples and defensive mechanisms like adversarial regularization or input transformations, can enhance robustness against such attacks.

c) Distribution Shifts:

- Distribution shifts occur when the input data during testing differs significantly from the data used during training.
- Robust models can generalize well to new or unseen data distributions, ensuring consistent performance.
- Techniques like domain adaptation, transfer learning, or using unlabelled data can help improve robustness to distribution shifts.

d) Model Uncertainty:

- Robust models provide measures of uncertainty or confidence in their predictions, particularly when faced with challenging or ambiguous inputs.
- Bayesian approaches, ensemble methods, or uncertainty estimation techniques, such as Monte Carlo Dropout or Variational Inference, can help quantify and incorporate model uncertainty into predictions.

e) Robust Feature Representation:

- Robust machine learning models are built on invariant features or are less sensitive to irrelevant variations in the input.
- Feature engineering techniques, such as normalization, scaling, dimensionality reduction, or robust feature extraction algorithms, can enhance the stability and resilience of the model.

f) Robust Training Techniques:

-Robustness can be enhanced through specific training techniques that mitigate the impact of noisy or adversarial data.

-Regularization techniques, such as L1 or L2 regularization, dropout, or early stopping, can help prevent overfitting and improve generalization performance.

g) Robust Evaluation Metrics:

Robustness can be assessed using evaluation metrics that capture the model's performance under challenging conditions.

Metrics like robust accuracy, adversarial robustness metrics (e.g., robust accuracy under adversarial attacks), or metrics that account for performance on different data distributions can provide a comprehensive understanding of the model's robustness.

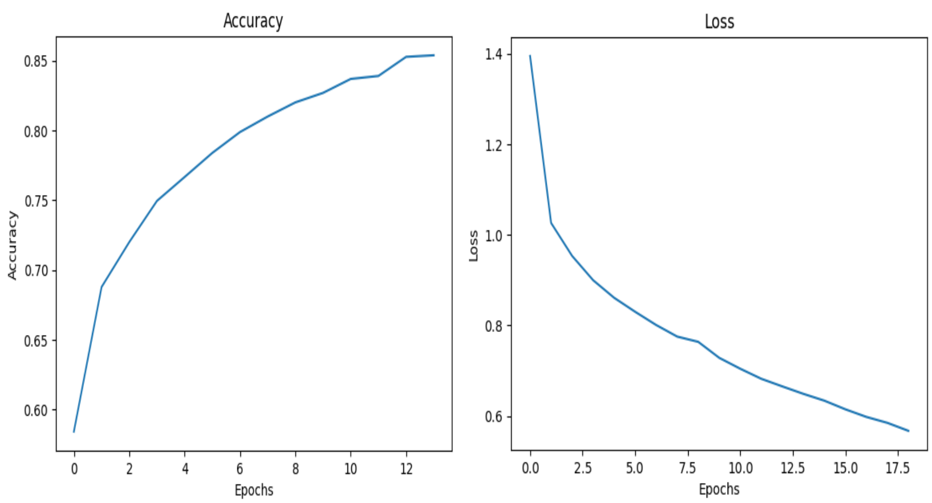
It is important to note that achieving complete robustness in machine learning models is often challenging, and there are trade-offs between robustness and other performance metrics, such as accuracy or complexity. Balancing these trade-offs and selecting appropriate techniques depend on the specific application domain, the nature of potential challenges, and the desired level of robustness required for the system.

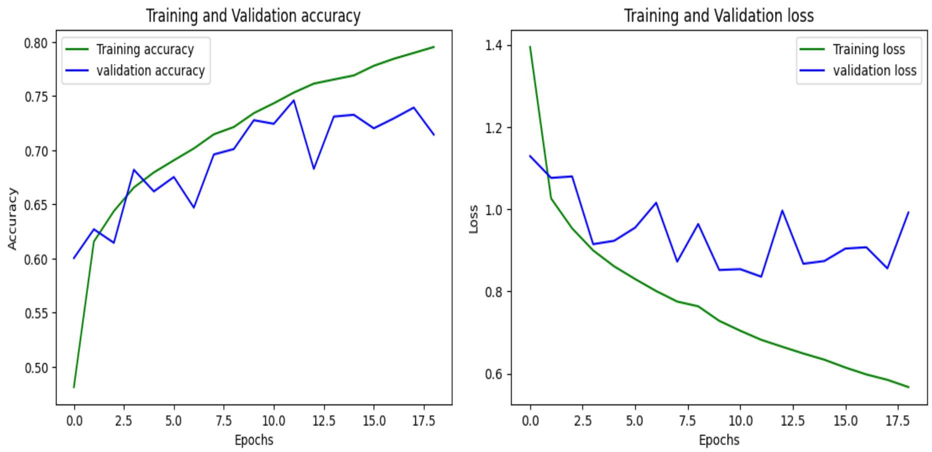
11 Convolutional Neural Networks

Convolutional Neural Network (CNN) is a deep learning algorithm well-suited for image recognition and processing tasks. It comprises multiple layers, including convolutional layers, pooling layers, and fully connected layers.

Hyperparameters used in CNN:

- Early Stopping: patience = 5, mode = max, verbose = 1
- No. of layers: 3
- Layers:
- Convolution layer: filters = 64(layer1), 32(Layer 2 and layer 3), strides = 2, kernel = 3,
- Batch Normalization
- Activation: RelU
- MaxPool Layer: strides = 2
- Regularization: L2
- Optimizer: Adam
- Loss: Categorical Cross entropy





12. Conclusion

After applying different types of classifiers including ensemble learning, our model has given the best performance in Extra Trees Classifier with accuracy of 91 percent and with ROC AUC score of 0.989. For AI Explainability we have used LIME, SHAP, and GRAD-CAM.

13. References

1 Srinilta, C., Kanharattanachai, S. (2019). Municipal Solid Waste Segregation with CNN. 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST).
2 J. Sousa, Ana Rebelo, A., Cardoso, J. S. (2019). Automation of Waste Sorting with Deep Learning. 2019 XV Workshop de Vis˜ao Computacional (WVC).
3 Lu, G., Wang, Y., Yang, H., Zou, J. (2020). One-dimensional convolutional neural networks for acoustic waste sorting. Journal of Cleaner Production, 122393.
4 Shah, A. V., Srivastava, V. K., Mohanty, S. S., Varjani, S. (2021). Municipal solid waste as a sustainable resource for energy production: State-of-the-art review. Journal of Environmental Chemical Engineering, 9(4), 105717. doi:10.1016/j.jece.2021.105717
5 Wang, C., Qin, J., Qu, C., Ran, X., Liu, C., Chen, B. (2021). A smart municipal waste management system based on deep learning and Internet of Things. Waste Management, 135, 20–29. doi:10.1016/j.wasman.2021.08.028
6 Waste Segregation Image Dataset,2022. [Online]. Available:

https://www.kaggle.com/datasets/aashidutt3/waste-segregation-imagedataset