RMIT International University Vietnam

Assignment Cover Page

Subject Code	COSC2753
Subject Name	Machine Learning
Location	SGS Campus
Title of Assignment	Group Machine Learning Project
Teacher Name	Dr. Bao Nguyen
Group Name	S1_G7
Group Members	s3974735 – Tran Manh Cuong s3978598 – Phan Nhat Minh s3986287 – Nguyen Vinh Gia Bao s3979654 – Le Viet Bao
Assignment Due Date	May 16, 2025
Date of Submission	May 16, 2025
Declaration of Authorship	We declare that in submitting all work for this assessment, we have read, understood and agreed to the content and expectations of Assessment Declaration as specified in https://www.rmit.edu.au/students/student-essentials/rights-and-responsibilities/
Consent to Use	We give RMIT University permission to use our work as an exemplar and for showcase/exhibition display.

Table of Contents

1. Introduction

Rice (*Oryza sativa*) is a staple food for more than half of the world's population and is central to the economies of many countries, especially in Asia (Kaggle 2022). However, rice crops are highly vulnerable to a wide range of diseases that can significantly reduce both yield and quality. Timely and accurate identification of these diseases is essential for effective management and to minimize economic losses. Traditionally, disease diagnosis has relied on expert visual inspection - a process that is often labor-intensive, slow, and inaccessible to farmers in rural or underserved areas.

The emergence of machine learning (ML) and deep learning (DL) technologies has transformed agricultural diagnostics by enabling automated, scalable solutions based on image data. Convolutional Neural Networks (CNNs), in particular, have proven highly effective in detecting and classifying plant diseases. For example, Deng et al. (2021) achieved 91% accuracy in diagnosing six rice diseases using an ensemble model combining DenseNet-121, SE-ResNet-50, and ResNeSt-50. Similarly, Haque et al. (2022) employed YOLOv5 for real-time detection of rice leaf diseases, reporting a mean average precision of 76% and an F1 score of 81%.

Several recent works have specifically leveraged the similar *Paddy Doctor* dataset to explore rice disease classification. Fatima (2023) implemented a CNN-based model trained from scratch, reaching over 82% validation accuracy. Abdullah (2024) applied transfer learning with DenseNet121, achieving 95% test accuracy through fine-tuning a lightweight classifier. Nadeem (2024) built a custom CNN using TensorFlow, incorporating rich metadata-driven exploratory analysis and attaining 80.5% validation accuracy. These studies demonstrate the feasibility and performance variation across modeling strategies, while also highlighting persistent challenges such as limited dataset diversity, inconsistent image quality, and similar visual symptoms among diseases.

In this project, we propose an end-to-end ML system that addresses three core tasks: (1) disease classification to determine plant health and identify specific ailments, (2) variety identification based on visual features, and (3) age prediction of the paddy plant in days. The system will be trained on a dataset of 10,407 labeled images across ten classes (nine diseases and one healthy category), each enriched with metadata on plant variety and age. Our workflow includes preprocessing, model training and selection, performance evaluation, and consideration of deployment strategies tailored for practical agricultural use.

2. Exploratory Data Analysis & Data Preprocessing

2.1. Exploratory Data Analysis

The dataset comprises 10,407 RGB images of paddy plants, each labeled as either "normal" or belonging to one of nine disease categories. Accompanying metadata includes the disease label, paddy variety, and age (in days). The dataset spans 10 rice varieties and 18 discrete age values ranging from 45 to 80 days.

To understand the dataset structure and identify potential biases, we visualized the distribution of labels, varieties, and ages using histograms and box plots. The analysis revealed class imbalance across both the label and variety attributes, with certain disease types and varieties being significantly underrepresented. The age distribution was more balanced, with most samples concentrated between 60 and 70 days and no extreme outliers. These findings informed several preprocessing decisions to ensure model robustness across all three prediction tasks.

2.2. Data Preprocessing

Preprocessing steps were critical to standardize the input data, enhance model training stability, and address quality issues in the raw images. The following procedures were applied:

- **Image Resizing & Cropping**: All images were resized to fixed dimensions for consistency - 256×256 pixels for Task 1, and 128×128 for Tasks 2 and 3, considering hardware constraints and deployment feasibility. Before resizing, center-cropping was applied to preserve aspect ratios and reduce geometric distortion.
- **Image Normalization**: Pixel values were scaled to the [0, 1] range using a "Rescaling" layer within each model architecture. This normalization ensures stable gradient updates and convergence during training.
- **Data Cleaning**: Corrupted or unreadable image files were removed. Grayscale images were excluded to maintain uniformity, as models were designed to accept three-channel RGB inputs.
- **Transparency Handling**: Images with alpha channels were converted to standard RGB by discarding the transparency layer, ensuring all inputs conformed to the expected dimensional structure.
- **Class Balancing & Resampling**: Even though models trained on imbalanced datasets may exhibit a high accuracy rate, they are often ineffective at identifying and correctly classifying instances of the minority class. If class imbalance isn't taken care of, it will result in the loss of important insights and patterns present in the minority class, leading to missed opportunities or critical errors. To address this, the training set was balanced by upsampling the underrepresented ones

to match the number of samples in the most common feature. This was done using random resampling with replacement, which helped mitigate the biases models can have toward majority classes.

Collectively, these preprocessing steps ensured the training data was clean, consistent, and balanced, which minimize bias and improve the generalizability of the models across real-world agricultural scenarios.

3. Modeling Approach & Evaluation

To solve the three tasks - disease classification, variety identification, and age prediction - we implemented and evaluated three deep learning models of increasing complexity:

1. **Artificial Neural Network (ANN)** – a fully connected baseline model,
2. **Deep Neural Network (DNN)** – a deeper version with regularization layers,
3. **Convolutional Neural Network (CNN)** – designed to leverage spatial hierarchies in image data.

All models were trained from scratch on resized RGB images (128×128 for Tasks 2 and 3; 256×256 for Task 1), using upsampled datasets to mitigate class imbalance. Transfer learning was excluded due to resource constraints, and early stopping was employed to prevent overfitting.

3.1. Task 1: Disease Classification

As detailed in the "Dataset Description & Preprocessing" section, we applied upsampling to address class imbalance and support generalization. Although the dataset was moderate in size (~16,000 samples), we opted against transfer learning due to significantly increased training time per epoch on available hardware. All models were trained from scratch.

Model	Input	Key Layers	Output	Justification
ANN	256×256×3 image	Dense(256, ReLU) ×3 → Dense(256, Sigmoid)	Dense(10, Softmax)	Baseline model using fully connected layers; lacks spatial feature understanding.
DNN	256×256×3 image	Dense(1024 → 512 → 256), each with BatchNorm and Dropout(0.3)	Dense(10, Softmax)	Deeper architecture with regularization; captures complex patterns but ignores spatial layout.
CNN	256×256×3 image	Conv2D(32 → 64 → 128 → 256), each with MaxPooling → Flatten → Dropout(0.2) → Dense(256, ReLU)	Dense(10, Softmax)	Learns local textures and spatial features important for disease recognition.

Each model trained for 45–60 minutes with early stopping. ANN showed gradual improvement but was hindered by a very small learning rate (1e-6), slowing convergence. The DNN performed poorly (accuracy < 0.5), likely due to limited data and lack of spatial feature learning. CNN significantly outperformed both, demonstrating strong spatial learning and class generalization (See figure A1 and figure A2).

Model	Val Accuracy	Val Loss	F1-score	Verdict
CNN	96.23%	0.1616	~0.9622	Delivered the highest accuracy by effectively learning spatial patterns and visual textures critical for disease identification.
ANN	76.96%	0.9358	~0.7672	Achieved moderate performance but was limited by its reliance on flattened inputs, reducing spatial feature retention.
DNN	43.34%	1.7354	~0.4287	Performed poorly due to its disregard for spatial hierarchies in images, leading to weak feature representation.

Overall, the CNN proved to be the most reliable model, achieving strong generalization across classes and consistently capturing key visual cues in disease patterns, making it well-suited for practical use in paddy disease detection systems.

3.2. Task 2: Variety Identification

For variety identification, the **CNN model was ultimately chosen** as the best-performing model. Its architectural design - stacked convolutional and pooling layers followed by dense classification layers - makes it ideal for capturing local textures and spatial hierarchies in plant images. Unlike the ANN and DNN, which rely on flattened pixel intensities, the CNN learns edge patterns and region-specific features crucial for visual classification of rice varieties. Its superior performance across accuracy, ROC-AUC, and classification metrics justified its selection.

Model	Input	Key Layers	Output	Justification
ANN	128×128×3 image	Dense(128, ReLU) ×3 → Dense(128, Sigmoid)	Dense(10, Softmax)	Simple baseline; lacks spatial understanding. Sigmoid added to introduce non-linearity before output.
DNN	128×128×3 image	Dense(1024 → 512 → 256 → 128), each with BatchNorm and Dropout(0.3)	Dense(10, Softmax)	Large hidden layers to capture nonlinear patterns. BatchNorm stabilizes learning; Dropout reduces overfitting.
CNN	128×128×3 image	Conv2D(32 → 64 → 128), each with MaxPooling → Flatten → Dropout(0.2) → Dense(128, ReLU)	Dense(10, Softmax)	Convolutions learn local texture and shape. Pooling down samples while preserving semantics. Dropout guards against overfitting.

One major challenge was **visual similarity across different paddy varieties**, such as overlapping leaf structures and color tones. This issue was compounded by natural variability in lighting, image angle, and resolution across the dataset. Additionally, class imbalance was present, requiring **upsampling of minority classes** to ensure fair learning across all ten varieties. The CNN model addressed these challenges through its ability to extract robust, hierarchical features, while data preprocessing (resizing, cropping, normalization) and label balancing improved model reliability (See figure B1 and figure B2).

Each model was trained and validated on an 80/20 split of the upsampled dataset (~70,000 images). Performance was measured using categorical accuracy, precision, recall, F1-score, ROC-AUC curves, and confusion matrices.

Model	Val Accuracy	Val Loss	F1-score	Verdict
CNN	99.66%	0.0165	~0.9965	High generalization, minimal overfitting, consistent predictions across all classes.
DNN	95.75%	0.1311	~ 0.9565	Strong overall but weaker than CNN for fine-grained visual distinctions.
ANN	92.23%	0.3488	~0.9180	Struggled to capture spatial patterns, resulting in lower performance, especially on similar-looking varieties.

Overall, the CNN model demonstrated exceptional classification performance and robustness, making it the recommended model for real-world deployment in identifying paddy plant varieties from image data.

3.3. Task 3: Age Prediction

This task aimed to estimate paddy plant age (in days) using RGB leaf images. Given the discrete labels (18 classes: 45–82 days), we treated this as a **multi-class classification** problem rather than regression. This improved stability, interpretability, and alignment with model evaluation techniques.

We again compared ANN, DNN, and CNN models. All were trained on ~55,000 balanced samples, preprocessed through center-cropping, resizing to 128×128, and normalization. The CNN achieved the highest performance by learning visual maturity cues such as leaf spread, curvature, and texture.

Model	Input	Key Layers	Output	Justification
ANN	128×128×3 image	Dense(128, ReLU) ×4	Dense(18, Softmax)	Baseline using fully connected layers without spatial awareness.

DNN	128×128×3 image	Dense(512 → 256 → 128 → 64), each with BatchNorm + Dropout(0.3)	Dense(18, Softmax)	Deeper architecture with regularization but no spatial feature extraction.
CNN	128×128×3 image	Conv2D(32 → 64 → 128), each with MaxPooling → Flatten → Dropout(0.2) → Dense(128)	Dense(18, Softmax)	Learns spatial maturity cues such as leaf spread and vein patterns.

After training, models were evaluated on an 80/20 validation split using categorical accuracy, precision, recall, F1-score, ROC-AUC curves, and confusion matrices. The CNN model consistently outperformed the other models, particularly in its ability to correctly identify neighboring age classes with minimal confusion (See figure C1 and figure C2).

Model	Val Accuracy	Val Loss	F1-score	Verdict
CNN	99.45%	0.0279	~0.9944	Best overall performance with excellent generalization and minimal loss.
ANN	95.92%	0.1570	~0.9578	Strong baseline but lacks spatial awareness.
DNN	83.81%	0.5229	~0.8206	Weak generalization and struggled with close-age classes.

Although all models demonstrated strong performance, the CNN stood out for its ability to capture fine-grained visual distinctions related to plant maturity, such as leaf spread, vein patterns, and curvature. Errors were typically limited to adjacent age classes (e.g., 70 vs. 72), indicating that the model had learned a meaningful biological representation of age.

4. Ultimate Judgement

Following comprehensive evaluation across all three tasks - disease classification, variety identification, and age prediction - we identified Convolutional Neural Networks (CNNs) as the most effective model architecture. This conclusion is based on robust analysis of model accuracy, generalization ability, and performance consistency across varied inputs.

CNNs consistently outperformed Artificial Neural Networks (ANNs) and Deep Neural Networks (DNNs), achieving validation accuracies above 96% and macro-averaged F1-scores exceeding 0.96 across all tasks. Their superior performance is attributed to their capacity to learn spatial hierarchies and local patterns essential for visual tasks like plant disease and variety recognition.

Compared to similar public efforts on the Paddy Doctor dataset, our CNN models achieved comparable or superior results. For example, Fatima (2023) trained a basic CNN from scratch and reported ~82% validation accuracy for disease classification. Nadeem (2024) also developed a from-scratch CNN using TensorFlow, achieving 80.5% validation accuracy. Abdullah (2024), who used transfer learning with DenseNet121, reached 95% test accuracy, but at the cost of greater training time and hardware requirements. In contrast, our models achieved 96.23% accuracy on disease classification and >99% on both variety identification and age prediction - without relying on transfer learning. This demonstrates that, with sufficient data balancing and tailored architecture design, high-performing models can be developed from scratch using lightweight CNNs.

Our CNNs also showed strong:

- **Generalization**: Robust across class imbalances and visually similar categories (e.g., age 66 vs. 68 or variety Surya vs. Ponni).
- **Resilience to Variability**: Maintained high performance despite noise in image resolution and lighting, supported by preprocessing and dropout regularization.
- **Interpretability Potential**: Architecture supports future integration of saliency maps or Grad-CAM to visualize decision rationale, aiding explainability in real-world use.

Task	Best Model	Validation Accuracy	F1-Score (Macro)	Verdict
Disease Classification	CNN	96.23%	~0.9622	Best spatial feature extraction

Variety Identification	CNN	99.66%	~0.9965	Strong visual separation by CNN layers
Age Prediction	CNN	99.45%	~0.9944	Accurate and biologically aligned

While our results are promising, further refinement could enhance real-world readiness:
- **Data Augmentation**: Including rotation, zoom, or lighting variation to simulate field conditions.
- **Transfer Learning**: Future work could reintroduce pretrained models (e.g., MobileNet, ResNet) for benchmarking and mobile deployment, as done in Abdullah (2024).
- **Model Compression**: Converting models to TensorFlow Lite or ONNX for integration with mobile devices used in agricultural settings.
- **Explainability Tools**: Incorporating Grad-CAM or LIME would enable agronomists to trust and interpret predictions.

In summary, our CNN-based models achieved state-of-the-art performance relative to public baselines, while maintaining training efficiency and architectural simplicity. Their superior ability to capture visual nuance across multiple tasks makes them highly suitable for integration into real-world agricultural diagnostics and decision-support systems.

5. Real-world Application

To validate the practical utility of our ML models, we developed and deployed a full-stack web application that enables real-time paddy leaf analysis for farmers. The system integrates our trained models into a RESTful API, which is used to process uploaded images and return predictions for disease classification, variety identification, and age prediction.

The frontend is a responsive web interface designed with a user-centric approach. Farmers can simply upload an image of a paddy leaf using a mobile phone or desktop browser. No additional input or manual preprocessing is required - the system automatically handles image normalization, model inference, and result display. The interface is optimized for mobile use, ensuring accessibility in rural or field environments where smartphones are often the primary device.

To enhance usability and trust, the application also provides a historical view, allowing users to browse past uploads and corresponding predictions submitted by others. This feature supports peer learning and can serve as a visual reference for identifying common diseases and comparing similar cases.

The entire system is deployed online (see figure D1), making it accessible from anywhere with an internet connection. This design removes the need for local installations or technical support, making it scalable and easily adoptable in real-world farming scenarios. By automating early disease detection and variety verification, the system has the potential to reduce reliance on manual inspections, support timely interventions, and ultimately improve yield outcomes.

Website URL: https://rice-plant-disease-classification.vercel.app/
API Documentation: https://rice-plant-disease-classification.k-clowd.top/docs

6. Conclusion

This project demonstrated the design, implementation, and deployment of an end-to-end machine learning system for paddy plant analysis, targeting three critical tasks: disease classification, variety identification, and age prediction. Through a comparative evaluation of ANN, DNN, and CNN architectures, we found that Convolutional Neural Networks consistently delivered the best performance across all tasks - achieving high accuracy, strong generalization, and reliable predictions even in the presence of visual variability and class imbalance.

Our CNN models were trained from scratch, without transfer learning, and still matched or outperformed several existing public implementations on the same dataset. This validates the effectiveness of tailored lightweight architectures when combined with rigorous preprocessing, balancing, and regularization techniques.

To ensure practical impact, we developed a user-friendly, mobile-responsive web application that integrates the trained models through a RESTful API. This solution enables farmers to upload images and receive instant diagnostic results from any internet-connected device, promoting early detection, peer learning, and informed decision-making in real-world agricultural settings.

Overall, our system not only meets the technical requirements of machine learning-based plant analysis but also addresses key usability and deployment challenges, making it a promising tool for scalable, accessible, and automated support in rice farming and crop management.

7. References

Abdullah, M. (2024). *Paddy Disease | Clean Solution*. Kaggle. https://www.kaggle.com/code/abdmental01/paddy-disease-clean-solution

Deng, R., Tao, M., Xing, H., Yang, X., Liu, C., Liao, K., & Qi, L. (2021). Automatic Diagnosis of Rice Diseases Using Deep Learning. Frontiers in Plant Science, 12, 701038. https://doi.org/10.3389/fpls.2021.701038

Fatima, S. (2023). *Rice Disease Classification Step-by-Step*. Kaggle. https://www.kaggle.com/code/samanfatima7/rice-disease-classification-step-by-step

Haque, M. E., Rahman, A., Junaeid, I., Hoque, S. U., & Paul, M. (2022). Rice Leaf Disease Classification and Detection Using YOLOv5. arXiv preprint arXiv:2209.01579. https://arxiv.org/abs/2209.01579

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep Learning in Agriculture: A Survey. Computers and Electronics in Agriculture, 147, 70–90. https://doi.org/10.1016/j.compag.2018.02.016

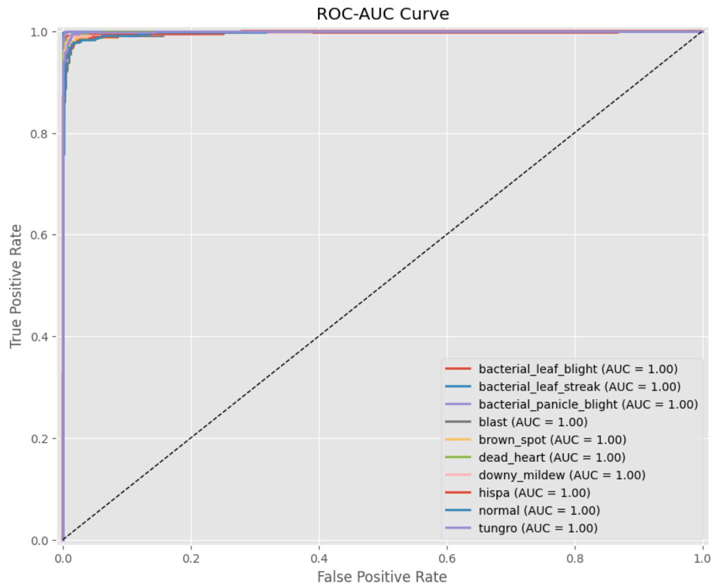
Kaggle. (2022). *Paddy Doctor: Paddy disease classification*. https://www.kaggle.com/competitions/paddy-disease-classification/overview

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science, 7, 1419. https://doi.org/10.3389/fpls.2016.01419

Nadeem, M. (2024). *Paddy Disease Classification Using CNN with TF*. Kaggle. https://www.kaggle.com/code/marianadeem755/paddy-disease-classification-using-cnn-with-tf

8. Appendices

Appendix A: ROC-AUC Curve and Confusion Matrix for the CNN Model in Task 1



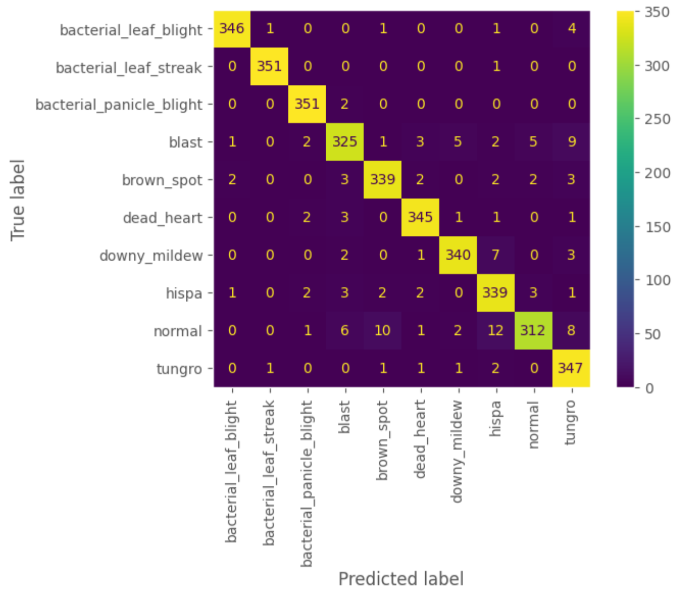
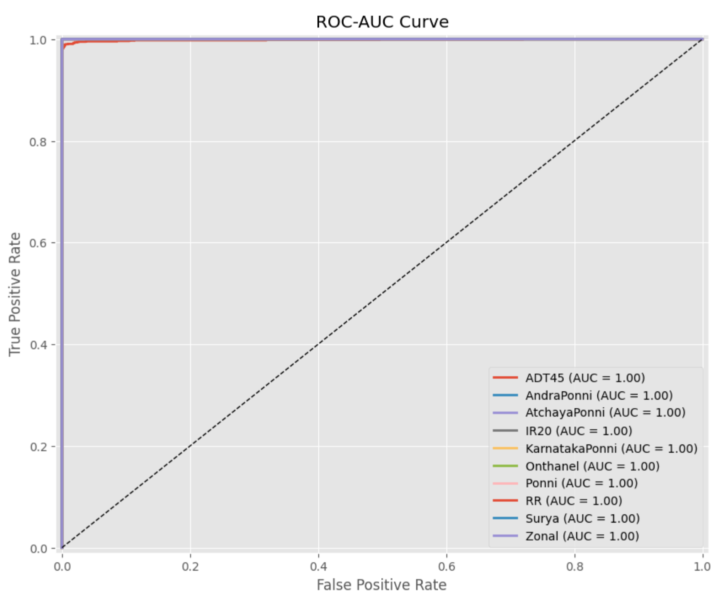


Figure A1. ROC-AUC Curve for the CNN Model in Task 1

Figure A2. Confusion Matrix for the CNN Model in Task 1

Appendix B: ROC-AUC Curve and Confusion Matrix for the CNN Model in Task 2



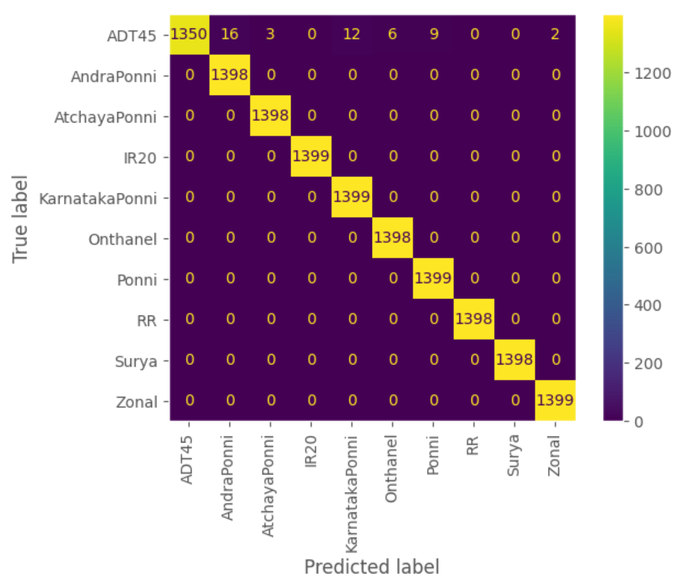


Figure B2. ROC-AUC Curve for the CNN Model in Task 2

Figure B2. Confusion Matrix for the CNN Model in Task 2

Appendix C: ROC-AUC Curve and Confusion Matrix for the CNN Model in Task 3

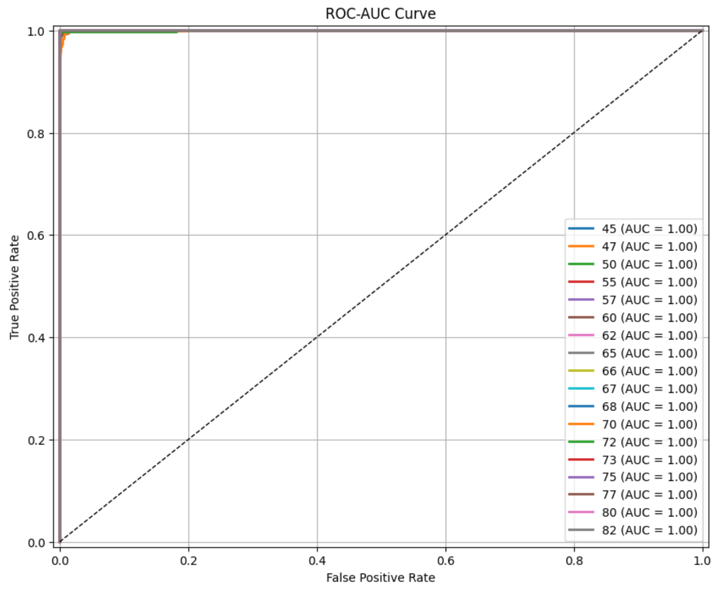


Figure C1. ROC-AUC Curve for the CNN Model in Task 3

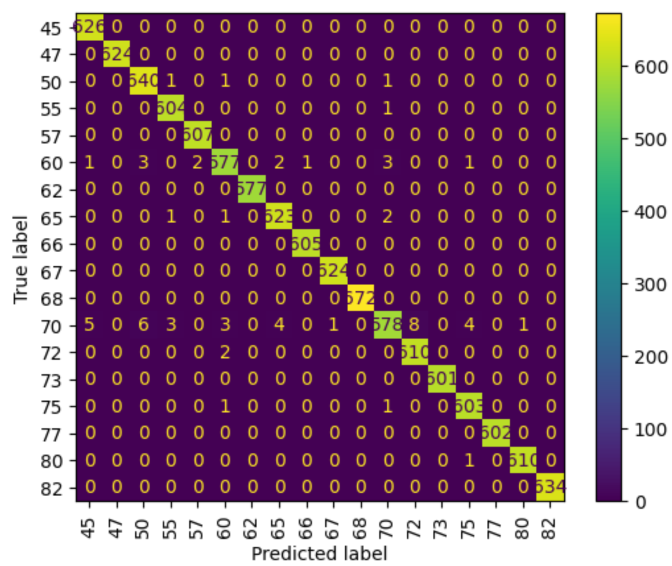


Figure C2. Confusion Matrix for the CNN Model in Task 3

Appendix D: Real-world Application Utilized Trained Machine Learning Models

Figure D1. Web Application Utilized Trained Machine Learning Models