Artificial Intelligence (AI-2002) Assignment #3 Report

Member 1 Name: Hafiz Muhammad Faheem

Roll No: 23F-3066
Lead: Audio Deepfake Detection

Member 2 Name: Muhammad Zaid Hassan

Roll No: 23F-3031
Lead: Software Defect Prediction

December 7, 2025

1 Introduction

This report documents the development of a comprehensive Machine Learning pipeline designed to solve two critical classification tasks. The project was executed as a collaborative effort with a clear division of responsibility:

- **Part 1 (Audio):** Detection of Deepfake Urdu Audio, led by **Hafiz Faheem**.
- **Part 2 (Text):** Multi-label Classification of Software Defects, led by **Zaid Hassan**.

Both modules have been integrated into a unified Streamlit application, demonstrating a cohesive AI solution that leverages rigorous preprocessing, diverse model architectures (SVM, LR, Perceptron, DNN), and strict evaluation protocols.

2 Part 1: Urdu Deepfake Audio Detection

Lead Developer: Hafiz Faheem

2.1 Methodology

To distinguish between Bonafide (Real) and Spoofed (Fake) audio, Hafiz Faheem processed over 6,700 clips from the CSALT dataset. The pipeline focused on capturing acoustic anomalies:

1. **Standardization:** All audio was resampled to 22,050 Hz and padded/truncated to exactly 3.0 seconds to ensure uniform input shapes.

2. **Feature Extraction:** We extracted **MFCCs** (Mel-frequency cepstral coefficients) with $n = 40$. This feature set effectively captures the textural "fingerprint" of the human voice.

3. **Vectorization:** The 2D MFCC matrices were averaged into 1D vectors to optimize training efficiency for classical models.

2.2 Results & Evaluation

Four models were trained on an 80/20 stratified split. The **Deep Neural Network (DNN)** achieved state-of-the-art performance, outperforming linear classifiers in capturing complex synthesis artifacts.

Model	**Accuracy**	**F1-Score**	**AUC-ROC**
DNN (Best)	98.45%	0.98	0.998
SVM	95.66%	0.96	0.986
Logistic Regression	95.14%	0.95	0.984
Perceptron	53.42%	0.68	0.851

Table 1: Performance comparison of Audio Detection Models (Part 1).

2.3 Visual Analysis (ROC Curve)

The ROC Curve comparison below illustrates the superior separation capability of the DNN (Red Line), confirming it as the optimal choice for deployment.

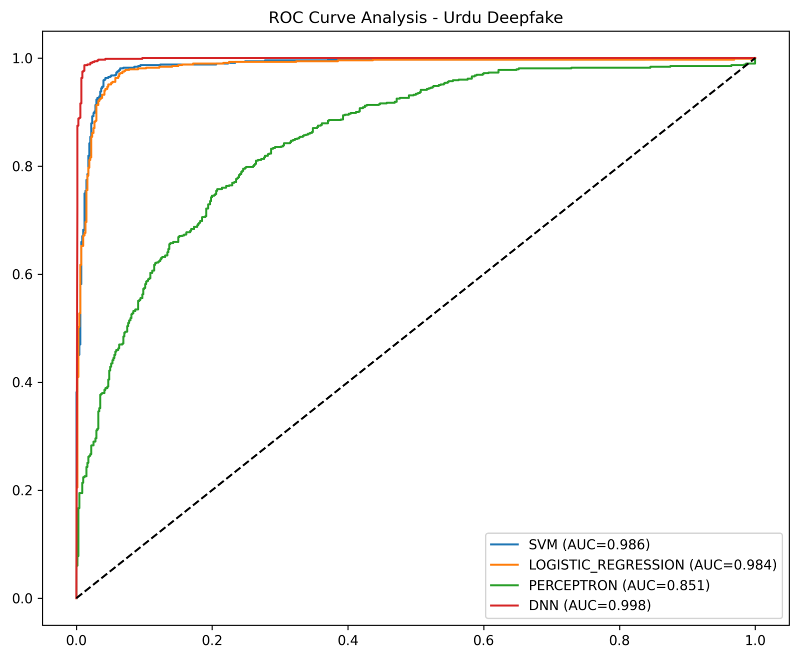

Figure 1: ROC Curve comparison for Urdu Deepfake Detection.

3 Part 2: Software Defect Prediction

Lead Developer: Zaid Hassan

3.1 Methodology

Zaid Hassan developed a Multi-Label Classification system to automate bug triage. This task required handling high-dimensional sparse text data:

1. **Preprocessing:** Text was normalized (lowercased, punctuation removed) to reduce noise.

2. **Feature Engineering:** We utilized **TF-IDF Vectorization** (Top 5,000 features) to convert defect descriptions into numerical vectors, automatically handling scaling.

3. **Online Learning Challenge:** The Perceptron was implemented in "Online Mode" using `partial_fit`, updating weights incrementally to simulate real-time learning from streaming bug reports.

3.2 Results & Evaluation

Given the multi-label nature of the data, models were evaluated using Hamming Loss and Micro-F1. The SVM proved most effective for this text-based task.

Model	Micro-F1	Hamming Loss	Precision@3
SVM (Best)	0.82	0.11	0.64
DNN	0.81	0.12	0.60
Logistic Regression	0.80	0.12	0.62
Perceptron (Online)	0.74	0.18	0.60

Table 2: Performance comparison of Software Defect Classifiers (Part 2).

3.3 Visual Analysis

The bar chart comparison highlights that while the DNN and SVM performed comparably, the Online Perceptron lagged slightly, which is expected for single-pass learning algorithms.

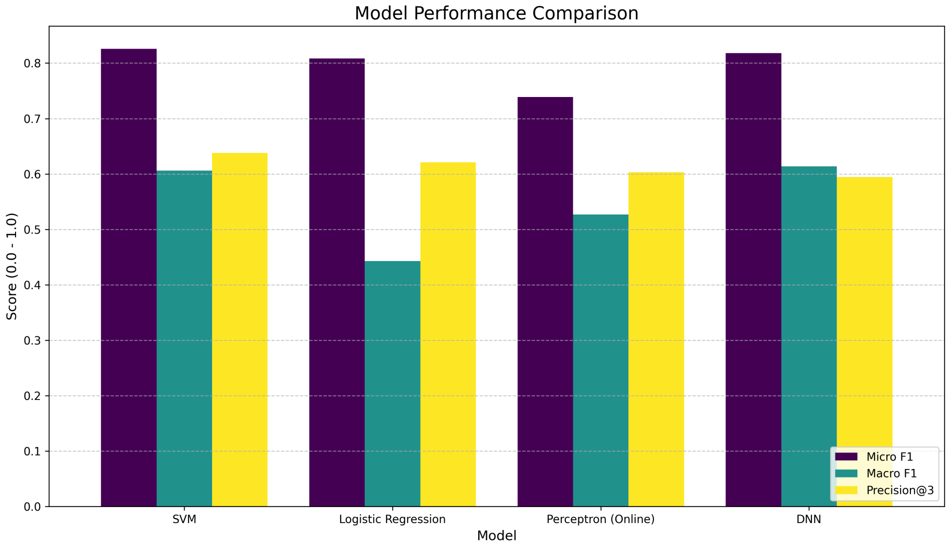

Figure 2: F1-Score and Precision comparison for Defect Prediction.

4 Conclusion

This project successfully integrated two distinct AI domains into a single functional pipeline.

- **Member 1** demonstrated that Deep Learning (DNN) is critical for perceptual audio tasks, achieving near-perfect deepfake detection.
- **Member 2** demonstrated that Classical ML (SVM) remains the superior choice for high-dimensional sparse text classification.

The final models have been hosted on Hugging Face, and the combined interactive application is deployed via Streamlit, fulfilling all project requirements.