National University of Computer and Emerging Sciences, Lahore

Deep Fake Audio Detection

Fayez Ali 21I-6228 BS(DS)

Zahra Hussain 21I-5615 BS(DS)

Syed Roohan Ali 21I-5629 BS(DS)

Supervisor: Dr. Hajra Waheed

Final Year Project

April 16, 2025

Anti-Plagiarism Declaration

This is to declare that the above publication was produced under the:

Title: Deep Fake Audio Detection

is the sole contribution of the author(s), and no part hereof has been reproduced as it is the basis (cut and paste) that can be considered Plagiarism. All referenced parts have been used to argue the idea and cited properly. I/We will be responsible and liable for any consequence if a violation of this declaration is determined.

Date: October 9, 2024

Name: Fayez Ali

Signature:

Fayez

Name: Zahra Hussain

Signature:

Zahra

Name: Syed Roohan Ali

Signature:

R Ali

Author’s Declaration

This states Authors’ declaration that the work presented in the report is their own, and has not been submitted/presented previously to any other institution or organization.

Abstract

This project focuses on developing a system for detecting deepfake audio in the Urdu language, which is particularly vulnerable to misuse in media and communication. Using a dataset of Urdu news recordings, we train and evaluate multiple models to distinguish between real and fake audio clips. Our goal is to ensure high accuracy in detection while reducing the time required for analysis. The system includes a user-friendly web interface where users can upload audio files for analysis. This project addresses the growing concerns over deepfake technology, offering a vital solution for mitigating audio-based scams and misinformation.

Executive Summary

This project aims to address the growing threat posed by deepfake audio, particularly in the Urdu language, by developing a robust detection system. Deepfakes have become a global concern as they have the capability of developing realistic audios that can be used for audio scams and misleading the public. Since Urdu is a language that is regarded as low resource, it has attracted very few researchers in this field hence exposing it. The primary objective of this project is to detect fake audio using advanced machine learning and deep learning techniques while providing users with a real-time detection solution.

The project is divided into two phases. In the first phase, we focus on model training and development. We use a dataset of news articles recorded in Urdu and apply various machine learning and deep learning techniques to classify audio clips as real or fake. First, the dataset is cleaned, addressing noise and inconsistencies, and standardized using libraries like Librosa and PyDub. Key audio features such as MFCCs, STFT, and Chroma are extracted for analysis. After feature extraction, the audio is labeled as either real or fake for supervised learning. The data is split into training, validation, and test sets, with normalization to improve model performance. Pre-trained models are fine-tuned, and custom deep learning architectures are designed to improve accuracy and reduce detection time.

In the second phase, a user-friendly web application is developed. This allows users to upload audio files and receive immediate feedback on the authenticity of the audio. The web app is designed for ease of use, with a simple interface. This Web app is intended for the general public, who may be at risk of scams involving deepfake audio, as well as professionals in media, cybersecurity, and fraud detection.

This report provides a detailed account of the project, including the problem statement, technical methodologies, and results. The literature review highlights prior work in audio deepfake detection, particularly in low-resource languages like Urdu, and how our approach differs. The methodology elaborates on the data preparation, feature extraction, and model development phases. Lastly, it includes the architecture of the project also functional requirements and future enhancements to improve the performance of system. Our project aims to contribute to the growing body of work on deepfake detection and to offer a practical tool that can help mitigate the risks associated with this technology.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

In recent years, the rise of deepfake technology has created significant challenges across various domains, from misinformation in media to privacy concerns in personal communications. Deepfake audio, which uses advanced techniques to create synthetic speech that mimics real human voices, is a major problem for content use especially in low-cost languages like Urdu when such technology grows and advanced effective methods is needed.The current lack of comprehensive solutions for Urdu highlights the need for research and development in this area. By using deep learning techniques, this project aims to create a reliable audio detection system in Urdu that can identify fake audio recordings immediately and accurately. This report includes an overview of the project vision and objectives, a detailed account of software and hardware requirements, an exploration of the high-level and low-level design of the system, and a comprehensive risk analysis. Each section aims to provide a clear understanding of the methods and considerations critical to the success of the deepfake audio detection project.

1.1 Purpose of this Document

The purpose of this document is to present the design, implementation, and evaluation of our project, which focuses on researching audio deepfake detection in the Urdu language. This project aims to address the gap in detecting deepfake audio in low-resource languages like Urdu. It gives a structured overview of the project's vision, comparison with other related works, and details the methodology, software requirements and high and low-level design.

The research question we aim to answer is: How can we develop an accurate, real-time deepfake audio detection system for the Urdu language using pre-trained and custom models? It aims to provide readers with a thorough resource to comprehend the project's importance, operation, and results.

1.2 Intended Audience

The intended audience for this project includes the general public, especially those concerned about fraud and the ethical implications of deep fake technologies in addition to researchers and students of artificial technology, and target industry professionals and students of engineering and cybersecurity. The project's objectives focused at collaboration and promoting knowledge sharing solutions for deep fake audio detection.

1.3 Definitions, Acronyms, and Abbreviations

1.3.1 Definitions

Automatic Speaker Verification (ASV): A method that verifies an individual's identity based on their voice, serving as a biometric recognition tool.

Artificial Intelligence (AI) : The capability of machines to perform tasks that typically require human intelligence, such as understanding language, recognizing patterns, and making decisions.

Text-to-Speech (TTS): A technology that transforms written text into spoken language, allowing for vocal output of text content.

Mel Frequency Cepstral Coefficients (MFCC): Features that are frequently used in speech and audio processing tasks to represent the short term power spectrum of an audio signal

Tacotron: A type of neural network that converts written text into speech by first creating a visual representation of the audio (spectrogram) and then generating the sound.

Variational Inference Text-to-Speech (VITS TTS): An advanced speech synthesis technique that combines deep learning approaches with traditional speech synthesis methods to produce natural-sounding voices.

Deepfake: Media—either audio or video—that has been artificially generated to mimic real indi- viduals, often used to create misleading representations.

Urdu Dataset: A carefully compiled collection of audio samples in Urdu, specifically designed for detecting deepfake content.

1.3.2 Acronyms and Abbreviations

MFCC : Mel Frequency Cepstral Coefficients

NLP : Natural Language Processing

ASV : Automatic Speaker Verification

AI: Artificial Intelligence

TTS : Text-to-Speech

CNN : Convolutional neural network

VITS TTS : Variational Inference Text-to-Speech

1.4 Conclusion

In summary, while AI technology has led to remarkable innovations, it has also introduced significant challenges, particularly in the form of audio deepfakes. These deepfakes can be exploited for malicious purposes, such as spreading misinformation and committing fraud. Although research in deepfake detection has mainly concentrated on widely spoken languages like English and Chinese, languages like Urdu have not received the focus they require. This project seeks to bridge that gap by developing arobust detection system for Urdu deepfake audio. By utilizing an advanced dataset and cutting-edge techniques, we aspire to contribute to ongoing research and future initiatives in processing low-resource languages.

Chapter 2 Project Vision

The project's vision is to build a powerful system that can detect fake audios using advanced machine learning and deep learning techniques, especially for the Urdu language. We aim to develop a tool that can quickly and accurately identify fake audio in less time. This will help improve the security and trustworthiness of audio products and prevent the misuse of deepfake technology. This chapter provides an overview of our project's problem statement and states the goals and scope of the project.

2.1 Problem Domain Overview

This paper discusses the rapid rise of deepfake technology, which has led to increasingly realistic fake audio content, posing significant challenges to security, fraud detection, and media integrity. While a good deal of the research has targeted audio deepfakes – in which a speaker's voice is mimicked—are increasingly being used to deceive people and establishments. Most deepfake detection research has centered on high-aid languages leaving an extensive hole for languages such as Urdu.This project aims to fill this gap by focusing on audio deepfake detection in Urdu, responding to a critical need for research and development in this under explored area.

2.2 Problem Statement

This addresses the rapidly evolving nature of deepfake audio technology and poses serious threats to individuals, organizations, and systems that rely on voice authentication. Current recognition methods mainly focus on feature-rich languages. Therefore, the challenge is to develop a system that can effectively detect deepfake audios in Urdu, using a dataset of news readings from native speakers, while achieving high accuracy.

2.3 Problem Elaboration

Deepfake audios exploit neural networks to generate synthetic voices that sound almost identical to real ones, making it difficult to differentiate between genuine and fake audio. The problem arises as we are working in Urdu, a low resource language, which have fewer datasets and models. We are using the dataset that consists of news reports read aloud by speakers. Fake audios are created using text-to-speech systems. Our major task is to build models that can differentiate between real or fake audios. This system should also be efficient enough for real time use. Additionally, need of web application is also important so that users can easily access our system for their ease. This will help them in identifying fake audios. This effort is critical in preventing the misuse of deep fake technology in sensitive contexts.

2.4 Goals and Objectives

The objectives and goals of this project are as follows:

- Train models capable of detecting real and fake audios in the Urdu language.
- Analyze and compare models of machine learning and deep learning and optimize it for our specific task.
- Optimize models to improve accuracy and real-time detection performance.
- Create simple and user-friendly web applications where users can upload and retrieve audio files and can get results.
- To support broader research on deepfake detection, especially for low-level features Language like Urdu.

2.5 Project Scope

The scope of this project is to develop a robust deep audio recognition system for Urdu language, focusing on classification accuracy and real-time capabilities. In the first phase, we will train our Urdu dataset using different customized pre-trained models. These models will be efficient and will be optimized for real time operation. Our results will be capable enough to detect fake or real audios. The second stage is the user- Creating friendly web application using HTML, CSS, Bootstrap, and a combination of Web-development frameworks like Python Flask. The pre-trained models will be integrated on the back-end. The user will upload audio file and will immediately get the result. A simple interface will be provided to the user for easy classification. This system will be available to people and organizations that will reduce the misuse of audios specially in Urdu language.

2.6 Sustainable Development Goal (SDG)

Figure 2.1: Sustainable Development Goals

This project aligns with SDG 16: ”Peace, Justice and Strong Institutions.” This project ensures the accurate detection of deep fake audios in low resource language i.e. Urdu. This helps in preventing the spread of misinformation and fraud activities, thus promoting peace and justice in digital world.

2.7 Constraints

- **Limited Dataset for Urdu**: One important limitation is that Urdu does not have large and diverse datasets for training. The dataset provided contains audio of news readings, which may limit the generalizability.The ability of the model to interact with other contexts such as casual conversation and lecture structure may differ.
- **Computational Resources**: Training deep learning models, especially those that aim for real-time performance, requires computational power. Resource limitations, such as access to high-performance GPUs, can affect the speed of model training and deployment.
- **Pre-trained Models**: Pre-trained models that are trained on high-resource language i.e. English, provide a good starting point. For Urdu, Fine-tuning these models and extracting additional features could present difficulties, specially in adapting them to the nuances of the language.
- **Language-Specific Features**: Urdu’s unique phonetics and difficult structure requires additional feature extraction techniques. The customisation in features is difficult for low resource language. Balancing the integration of language-specific features with generalizability makes the model design difficult.

2.8 Conclusion

This project addresses the vital issue of deepfake audio detection in Urdu, focusing on both high accuracy and real-time performance. In phase one, we will train and fine-tune models to classify real and fake audios, while phase two will focus on optimizing the models for real-time use and developing a user-friendly web app. Despite challenges like limited data and computational constraints, the successful completion of this project will strengthen the security and authenticity of voice communications in Urdu, contributing to the fight against fraud which aligns with UN Sustainable Development Goal (SDG) 16, promoting peace, justice, and strong institutions by ensuring the integrity of digital communications.

Chapter 3 Literature Review / Related Work

The following section covers a comprehensive literature review of previous research done concerning our project while also providing detailed insights and knowledge into how the concepts can be applied in our context. It also provides the applications relating to the project.

3.1 Definitions, Acronyms, and Abbreviations

3.1.1 Definitions

Artificial Intelligence: The creation of computer systems that are capable of carrying out operations that ordinarily require human intelligence, like speech recognition and decision-making, is referred to as artificial intelligence

Deepfake: A deepfake is a synthetic media technique that uses artificial intelligence (AI), particularly deep learning, to manipulate or generate audio, video, or images that mimic real people. It can convincingly alter appearances or voices, making it difficult to distinguish between real and fake content.

Deep Learning: A subset of machine learning that focuses on using deep neural networks with many layers to perform complex tasks.

Machine Learning Techniques: Machine learning techniques are strategies and algorithms that let computers learn from information and come to conclusions or predictions on their own without needing to be explicitly programmed.

Neural Network model: A computational framework for pattern recognition, classification, and regression tasks that draws inspiration from the structure and functions of the human brain

Audio Features: Pitch, speech rate, mean energy, mean intensity, wavelength, and energy are among the characteristics that are taken out of audio data and used in analysis and classification

Spectrogram: A visual representation of the spectrum of frequencies in a sound signal as it varies with time, commonly used in audio analysis to identify patterns and features in speech or music.

Speech Synthesis: A process of artificially generating human speech, commonly used in Text-to-Speech (TTS) systems, which can be exploited in deepfake audio generation.

Waveform: A graphical representation of a sound wave, showing how the amplitude of sound changes over time, often used in audio signal processing.

Feature Extraction: Extracting useful features(data) from the given dataset.

Audio Classification: The task of categorizing audio signals into predefined categories based on features extracted from the audio, such as distinguishing between real and fake audio.

3.1.2 Acronyms and Abbreviations

NLP: Natural Language Processing

MFCC: Mel-frequency cepstral coefficients

ASR: Automatic Speech Recognition

GAN: Generative Adversarial Network

SVM: Support Vector Machine

LSTM: Long Short-Term Memory

RNN: Recurrent Neural Network

DNN: Deep Neural Network

STFT: Short-Time Fourier Transform

CNN: Convolutional Neural Network

FAD: Fake Audio Detection

MFCC: Mel-frequency Cepstral Coefficients

SR: Sampling Rate

TTS: Text-to-Speech

VT: Vision Transformer

3.2 Detailed Literature Review

A lot of work has already been done in this field. However, no such research has been done in a low resource language like urdu. Many people research in this field and experiment with different models for audio deepfake detection. The study of such type of researches are important in effective implementation of our project. One of the major addition in urdu deepfake is the high quality dataset that has recently publish for public use but the detection part is still missing. So our aim in this project is not only to provide high accuracy detection but also provide a webapp where a user can upload an audio to check its authenticity. The literature review provides a comprehensive analysis of deep fake audio detection, security threats, audio data, deep learning models, and approaches used in detection by evaluating evaluated studies, research work, and related articles.

3.2.1 Related Research Work 1

3.2.1.1 Summary of the research item

Marium Mateen [1] discussed about detecting the audio deepfakes in the Urdu language. The dataset is collected from different sources like news, ted talks, informative videos and urdu lectures. Original audios were converted into fake audios using real time voice cloning applications. The dataset consists of 400 audio clips (real and fake) and aims at utilizing deep learning techniques to detect audio deepfakes. The researcher then use LSTM model to detect audio deepfakes produced using imitation and synthesis based techniques and get the accuracy of 91%

3.2.1.2 Critical analysis of the research item

Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks are also part of this research in trying to build a model that will classify if an audio is fake or not. LSTM architecture is Good in processing long dependencies but in case of detecting audio deepfakes we don't need to store the context because if at any point we find the fake generation we will consider it as fake audio. This paper applied only one model to detect the deepfakes. The dataset used is not enough and not of good quality. Also the deepfake conversion of original audios is not a proper way that the researcher chooses in this [1] paper.

3.2.1.3 Relationship to the proposed research work

Both our research and the reviewed research are focused on urdu deepfake detection. There is little like no work has been done in the Urdu language. In this way this paper helps us in understanding better implementation for urdu deepfake detection. Depending on the combination of RNN and LSTM, the present research offers a solution in detecting deepfakes more effectively catering towards the Urdu language.

3.2.2 Related Research Work 2

3.2.2.1 Summary of the research item

Saleem et al. [2] investigates the use of Cyclic Generative Adversarial Networks (Cyclic GANs) to address the issue of spoofed voice detection in Urdu and English speech verification systems. By employing a two-fold solution, the study first describes a one-to-one voice conversion method using Cyclic GANs that generate speech from a source speaker to a target voice bi-directionally. These recordings from the GAN produced model contained the female → female , female → male, male → male and male → female converted voices. Each set of recordings had 250 voice utterances and for the English they used a prebuilt dataset. It then uses adversarial examples produced during this process for spoofed voice detection. The methodology leverages the unique cyclic consistency loss of Cyclic GANs, ensuring close resemblance between the input and generated outputs, and employs Gradient Boosting to classify genuine and spoofed utterances effectively.

3.2.2.2 Critical analysis of the research item

While the study demonstrates promising results in detecting spoofed voices, the reliance on a substantial amount of training data and the complexity of the Cyclic GAN architecture could pose challenges for real-world applications. The research underscores the potential of using advanced machine learning techniques in automated speech verification systems but also highlights the need for more robust data sets and further experimentation to enhance the practicality and reliability of the proposed solutions.The dataset contains the single word utterances in urdu that is not good approach in making dataset for audio deepfakes. Further analysis into the model's behavior across diverse acoustic environments and speaker variations would be beneficial to assess its real-world efficacy.

3.2.2.3 Relationship to the proposed research work

Both the discussed study [2] and our research work focus on Urdu language for detecting manipulated audio content. The methodologies share a common goal of working on urdu language audio deepfakes detection.

3.2.3 Related Research Work 3

3.2.3.1 Summary of the research item

Valson et al. [3] examines the use of speech positioning algorithms to detect deep voices The study combines natural voice characteristics with speech positions as markers for identification of the difference between real and artificial voice. Using a dataset generated from 49 participating voice samples, the team used five machine learning algorithms. An ADA model emerged as the most capable of classification, achieving an accuracy of 81

3.2.3.2 Critical analysis of the research item

The method shows promise by incorporating sensitivity and physics into the detection system, potentially increasing the robustness of in-depth lie detection tools. However, the study faces limitations such as relying on sufficiently diverse datasets for effective model training due to complex analysis requirements of speech pause patterns and possible scalability issues. The exclusively English-focused experiments did not account for the potential impact of diverse accents or languages on our results

3.2.3.3 Relationship to the proposed research work

The focus on deepfake detection is consistent with the broader study and utility of protecting digital communications from emerging threats. The study of alternative approaches and findings can serve as insights for developing advanced audio depth programs. This unique approach will help us to get better accuracy because in AI generated voices there are no such types of natural properties.

3.2.4 Related Research Work 4

3.2.4.1 Summary of the research item

Pham et al. [4] introduce an improved deep learning algorithm for deepfake audio detection. The system uses a combination of spectrogram-based features generated by various transformations (STFT, CQT, WT) and earwax filters (Mel, Gammatone, LF, DCT) It monitors how patterns and advanced such as CNNs, RNNs, Whisper and Speech Brain improve audio deep learning architectures for audio deep recognition. It turns out that the proposed cluster model combines these methods and shows competitive performance at equal error rate (EER).) of 0.03 to be obtained on the ASVspoof 2019 data set.

3.2.4.2 Critical analysis of the research item

This study addresses the important challenge of deepfake audio recognition by combining innovative spectrogram-based and deep learning models The ensemble method helps to capture audio inconsistencies and it extends by integrating models However, the application of the research to real situations remains to be fully tested, and the complexity of the system may pose challenges in practice settings

3.2.4.3 Relationship to the proposed research work

Both this study [4] and the proposed work use deep learning techniques for media integrity, but this study mainly uses the combination of spectrogram transformation and sampling for audio fidelity, and focuses on the importance of a multidisciplinary approach in safety-focused implementation. Also the important part of this paper is the use of image based models on audio dataset by making spectrograms graphs type features so, this helps us in using advance image based models like Vision transformers

3.2.5 Related Research Work 5

3.2.5.1 Summary of the research item

Munir et al. [5] presents the development and evaluation of a new Urdu deepfake audio data set developed for training deepfake recognition models. The data set was generated using both Tacotron and VITS TTS text-to-speech (TTS) techniques to create deepfake audio. Studies using the AASIST-L model show equivalent error rates (EERs) of 0.495 and 0.524 for VITS TTS and Tacotron-generated audio, respectively, indicating moderate detectability Human studies were also conducted, where it was found that most people perceive deepfake audio struggles, with ROC curve analysis showing an area under the curve (AUC) of 0.63

3.2.5.2 Critical analysis of the research item

This paper contributes significantly to the field of deepfake detection by providing a specialized dataset in a low-resource language, Urdu. The study's reliance on specific TTS methods might limit its generalization across different spoofing technologies. Moreover, The dataset's reliance on a convenience sample leads to a gender imbalance in the speakers, highlighting the need for a more diverse dataset

3.2.5.3 Relationship to the proposed research work

We are using this dataset in our project as this is the only high quality dataset till yet made for urdu deepfake audio detection. In this paper the researchers only introduce the dataset and no detection models are applied. This still doesn't fill the gap of audio deepfake detection in urdu.So our project aims to focus on this research gap

3.2.6 Related Research Work 6

3.2.6.1 Summary of the research item

Hamza et al. [6] detects deepfake audio using machine learning techniques, focusing on Mel-frequency cepstral coefficients (MFCCs) as feature. Utilizing the Fake-or-Real dataset, the research applies various machine learning models, including Support Vector Machines (SVM) and VGG-16 deep learning models, achieving notable success in classifying deepfake from genuine audio. The methodology emphasizes the robustness of MFCC features in capturing essential audio characteristics that help distinguish between authentic and manipulated audio.

3.2.6.2 Critical analysis of the research item

The approach showcases the efficacy of using MFCC features combined with powerful machine learning algorithms to detect audio deepfakes. However, the adaptation of these models to diverse and possibly more complex real-world scenarios remains to be thoroughly tested. Additionally, while the VGG-16 model demonstrates superior performance, the computational demand and potential overfitting due to high model complexity could pose challenges in operational settings.

3.2.6.3 Relationship to the proposed research work

This study purely implemented machine learning models and helps us in understanding a deeper insight into the implementation of those models. Our proposed approach of using other significant features to detect will surely make us able to achieve better accuracy

3.2.7 Related Research Work 7

3.2.7.1 Summary of the research item

Zaynab Almutairi and Hebah Elgibreen [7] reviews various machine learning and deep learning methods developed to detect audio deepfakes (ADs). It categorizes audio deepfakes into synthetic-based, imitation-based, and replay-based types, and discusses the methodologies for detecting each type. The review highlights the trade-offs between accuracy and computational complexity in AD detection, and emphasizes the need for further research to address existing gaps in the field, such as detecting accented voices or audio in noisy environments.

3.2.7.2 Critical analysis of the research item

The paper provides a comprehensive overview of current AD detection techniques, presenting a balanced view of their strengths and limitations. It critically evaluates the performance impact of different method types and audio features, pointing out the significant challenge of generalization across diverse real-world scenarios. However, the review suggests that further advancements are necessary to enhance the robustness and adaptability of audio deepfake detection systems, particularly in non-English languages and under varied acoustic conditions.

3.2.7.3 Relationship to the proposed research work

Both the reviewed study and our project are concerned with the detection of fake audio content, although the reviewed work takes a broader approach by discussing a variety of detection methods and their applicability. The insights from this review could inform us by highlighting effective techniques and potential pitfalls in audio deepfake detection, encouraging a focus on innovation and improvement in detection algorithms.

3.2.8 Related Research Work 8

3.2.8.1 Summary of the research item

Mcuba et al. [8] looks at deepfake audio detection methods using deep learning, focusing on CNN architectures. It finds how these architectures handle different audio features such as MFCC, Mel-spectrum, Chromagram, and spectrogram for forensic applications. Research shows that custom CNN architectures optimized for specific features perform well, with VGG-16 excelling in handling MFCC features. The study is important for forensic scholars, providing insights into the reliability and validity of deep learning techniques for artificial voice recognition.

3.2.8.2 Critical analysis of the research item

The paper effectively bridges the gap between deep learning applications and forensic requirements, providing a detailed comparison of CNN frameworks for deep learning analysis but reliance on controlled data sets may affect findings integrating it all into a real-world setting. Furthermore, the study highlights the importance of continuous detection methods to improve the quality of the developed deepfake technology and shows that combining CNN algorithms can enhance their detection ability.

3.2.8.3 Relationship to the proposed research work

This study purely used image based audio features and applied deep learning models and helps us in understanding a deeper insight into the implementation of those models. Our proposed approach of using other significant features to detect will surely make us able to achieve better accuracy

3.2.9 Related Research Work 9

3.2.9.1 Summary of the research item

Yi et al. [9] presented a systematic overview of the developments in audio deepfake detection. It highlights the variations among different types of deepfake audio, examines competitions, datasets, features, classifications, and evaluates state-of-the-art methods. The study emphasizes the need for larger scale datasets, improved generalization across unknown fake attacks, and better interpretability of detection results.

3.2.9.2 Critical analysis of the research item

While this survey collates extensive information and presents a unified evaluation of various methods, it also points out significant gaps such as the lack of large-scale datasets for in-the-wild scenarios and the poor generalization of existing methods to new, unknown types of fake attacks. The survey serves as a vital resource for researchers by detailing current achievements and highlighting areas requiring further research and development.

3.2.9.3 Relationship to the proposed research work

The survey's focus on enhancing the generalizability and effectiveness of deepfake detection methods aligns with the broader goals of improving security against digital threats, similar to our project's aim to develop robust detection mechanisms for deepfake audio in Urdu.

3.2.10 Related Research Work 10

3.2.10.1 Summary of the research item

Muller et al. [10] examines the general applicability of audio deepfake detection systems. The study critically evaluates twelve architectures previously used in audio spoof detection, and reuses them to test these models beyond the traditional ASVspoof benchmark with a consistent evaluation standard to perform on another dataset with 37.9 hours of popular politicians audio, of which 17.2 hour is deepfakes.

3.2.10.2 Critical analysis of the research item

The study found significant performance degradation when models trained on conventional datasets were tested on this new, more realistic dataset, with performance deteriorating by up to 37%. This suggests that existing models are overly tailored to benchmarks and may not perform well in real-world scenarios. The research highlights the necessity for developing models that can operate effectively across diverse and unforeseen data environments .

3.2.10.3 Relationship to the proposed research work

This research underscores the importance of robust model evaluation and the potential pitfalls of overfitting to specific datasets. It relates to broader efforts in developing deepfake detection technologies that are effective in real-world applications, aligning with the goals of improving security and authenticity in digital media.

3.2.11 Related Research Work 11

3.2.11.1 Summary of the research item

Govindu et al. [11] at MIT World Peace University employs Generative Adversarial Neural Networks (GANs) and Explainable Artificial Intelligence (XAI) techniques to detect deepfake audio. They use the Fake or Real (FoR) dataset, assessing the generated audio quality with the Fréchet Audio Distance (FAD). Additionally, the study integrates XAI tools like LIME, SHAP, and GradCAM to provide insights into the decision-making process of the models used.

3.2.11.2 Critical analysis of the research item

This analysis helps activate the field by connecting the XAI, thereby increasing the transparency of the search process. The use of the FAD score results in quantifiable audio quality, but the research could benefit from greater validation across different data sets to increase its applicability in real-world settings.

3.2.11.3 Relationship to the proposed research work

This study's use of GANs and XAI for audio deepfake detection parallels the broader goal of enhancing security protocols in digital communications, similar to other projects aiming to improve deepfake detection mechanisms. The integration of XAI could provide a model for future research aiming to make AI decisions more interpretable and trustworthy.

3.2.12 Related Research Work 12

3.2.12.1 Summary of the research item

Mouna Rabhi, Spiridon Bakiras and Roberto Di Pietro [12] addresses the vulnerability of audio deepfake detection systems to adversarial attacks. They identify that conventional deepfake detectors, while effective under normal conditions, can be deceived by specially crafted adversarial inputs, reducing detection accuracy significantly. They propose a robust, lightweight defense mechanism that significantly mitigates the effects of these attacks.

3.2.12.2 Critical analysis of the research item

This research highlights a critical gap in the security of audio deepfake detectors by demonstrating their susceptibility to adversarial attacks, which can dramatically lower detection effectiveness. The proposed defense mechanism, while innovative, needs validation in diverse real-world scenarios to evaluate its practical effectiveness and adaptability.

3.2.12.3 Relationship to the proposed research work

The vulnerabilities and countermeasures identified in this study are directly relevant to the broader field of digital media authentication, underscoring the importance of enhancing security features in detection systems to handle evolving adversarial techniques

3.2.13 Related Research Work 13

3.2.13.1 Summary of the research item

Farkhund Iqbal and Abdul rehman [13] explores advanced machine learning methods to detect audio deepfakes. It highlights the use of optimal feature engineering and machine learning classifiers on a dataset designed to test fake or real audio. Techniques such as various feature extraction methods are employed to improve the accuracy and efficiency of detecting fraudulent audios.

3.2.13.2 Critical analysis of the research item

This study advances the field of audio deepfake detection by integrating comprehensive feature engineering with robust machine learning models, achieving an accuracy gain of 26 percent over baseline methods. However, the focus on a single dataset might limit the generalizability of the proposed methods to other types of audio or conditions not covered by the dataset.

3.2.13.3 Relationship to the proposed research work

The methodology and challenges addressed in this paper align with the broader objectives of enhancing digital security through better detection systems for fake media. This work's emphasis on feature engineering and specific classifier efficacy provides a valuable reference point for similar research in audio authenticity verification.

3.2.14 Related Research Work 14

3.2.14.1 Summary of the research item

Mathew et al. [14] explores the development of real-time deepfake audio detection systems for communication platforms, addressing the limitations of static models in dynamic audio streams. The research introduces an executable software that supports real-time deepfake detection and evaluates its performance using the ASVspoof 2019 dataset and a new dataset from actual communication sessions.

3.2.14.2 Critical analysis of the research item

The paper underscores the challenge of adapting static deepfake detection models to real-time scenarios, revealing a significant performance drop in dynamic conditions. It suggests enhancements through specialized datasets and model training strategies to better cope with the variations in real-world communication platforms.

3.2.14.3 Relationship to the proposed research work

This research [14] parallels the broader goals of ensuring audio integrity in digital communications by advancing real-time detection capabilities. It contributes to understanding the complexities of applying static models to dynamic environments, which is crucial for maintaining security against deepfake threats in real-time interactions.

3.2.15 Related Research Work 15

3.2.15.1 Summary of the research item

Jordan J. Bird and Ahmad Lotfi [15] developed a system for real-time detection of AI-generated speech, using the DEEP-VOICE dataset. This dataset comprises real human speech and converted speech from eight notable figures. They implemented hyperparameter optimization for machine learning models, notably the Extreme Gradient Boosting model, achieving a classification accuracy of 99.3 percent in real-time.

3.2.15.2 Critical analysis of the research item

The study presents a significant advancement in detecting deepfake audio, particularly in its ability to operate in real-time. However, its reliance on a specific dataset might limit the generalization of the findings. Future research could expand on dataset diversity and test the system's robustness across more varied audio conditions.

3.2.15.3 Relationship to the proposed research work

This research provides crucial insights into the real-time detection capabilities necessary for practical applications, aligning with broader efforts to secure digital communication against deepfake technologies.

3.2.16 Related Research Work 16

3.2.16.1 Summary of the research item

Xinfeng et al. [16] introduces SafeEar, a novel framework designed to detect audio deepfakes without exposing semantic content. It achieves this by decoupling semantic and acoustic information, utilizing only acoustic data for deepfake detection. This ensures privacy protection while maintaining high detection performance across multiple languages and audio conditions.

3.2.16.2 Critical analysis of the research item

The innovation in SafeEar addresses critical privacy concerns related to exposing sensitive audio content during deepfake detection. However, the effectiveness of decoupling acoustic and semantic elements and the framework's performance in extremely noisy or varied environments remain areas for potential improvement.

3.2.16.3 Relationship to the proposed research work

SafeEar's methodology offers significant implications for enhancing both the security and privacy of audio communications in multiple languages, aligning with global needs for robust deepfake detection systems that do not compromise confidential information.

3.2.17 Related Research Work 17

3.2.17.1 Summary of the research item

Zhang et al. [17] developed a continual learning approach named Radian Weight Modification (RWM) for detecting audio deepfakes. This method categorizes classes into compact and spread-out feature distributions and uses these distinctions to optimize learning paths dynamically, enhancing model adaptability and reducing forgetting when new fake audio types are encountered.

3.2.17.2 Critical analysis of the research item

RWM stands out by allowing dynamic adaptation to new data without needing prior data, suggesting potential for broad application in audio deepfake detection and other machine learning fields. However, its dependency on the distinctiveness of feature distributions may limit its effectiveness across less diverse datasets.

3.2.17.3 Relationship to the proposed research work

The study's exploration of continual learning techniques directly relates to efforts to improve the resilience of detection systems against evolving deepfake techniques, offering valuable insights into sustainable model training strategies.

3.2.18 Related Research Work 18

3.2.18.1 Summary of the research item

Yuang et al. [18] develops a new cross-domain audio deepfake detection (ADD) dataset using five advanced zero-shot text-to-speech (TTS) models, addressing the limitations of current datasets. They introduce novel attack-augmented training methods, evaluating model performance across different audio conditions to simulate real-world scenarios.

3.2.18.2 Critical analysis of the research item

The creation of a cross-domain dataset is a significant advancement for ADD, but the focus on zero-shot TTS models might limit the dataset's applicability to broader TTS technologies. Additionally, while the new training approaches show promise, their effectiveness in live environments remains to be thoroughly tested.

3.2.18.3 Relationship to the proposed research work

This study works on making a dataset for audio deepfakes detection using text to speech models. The dataset we use for our research is also based on text to speech models so, by studying the structure and creation of such models we will be able to better understand the nature of deepfake

3.2.19 Related Research Work 19

3.2.19.1 Summary of the research item

Anton Firc, Kamil Malinka and Petr Hanáček [19] explores the use of different spectrogram visualizations for deepfake audio detection. They analyze the performance, hardware requirements, and speed of various spectrogram types in deep neural network-based detection systems, finding no one-size-fits-all solution but rather a dependency on specific use-case requirements.

3.2.19.2 Critical analysis of the research item

The research provides valuable insights into the practical aspects of implementing deepfake detection systems, highlighting the trade-offs between accuracy and resource demands for different spectrogram types. However, it notes the challenges in selecting the most effective spectrogram without specific context, suggesting further research is needed to optimize detection systems for varied applications.

3.2.19.3 Relationship to the proposed research work

The in depth analysis of different spectrograms in this paper creates a clear understanding about them. In our project we will implement advanced image based models and use these spectrograms as input to those models. This paper helps us in implementing those models

3.2.20 Related Research Work 20

3.2.20.1 Summary of the research item

Valente et al. [20] developed a Convolutional Neural Network (CNN) model to detect deepfake audio using Mel spectrograms. The model was trained and tested across several voice datasets including FoR, ASV, and WaveFake, achieving high accuracy in detecting deepfakes, indicating the effectiveness of CNNs in differentiating genuine from fake audio.

3.2.20.2 Critical analysis of the research item

The study [20] demonstrates the CNN model's robustness in audio deepfake detection, validated by high accuracy across multiple datasets. However, challenges such as adapting to new deepfake methods and further enhancing model generalizability were noted.

3.2.20.3 Relationship to the proposed research work

The implementation of CNN model in this paper will help us in our project to implement this model with better understanding of the architecture.

3.3 Literature Review Summary Table

During the course of our research, we studied many existing papers published in the domain of audio detection. We observed that a range of techniques and methodologies were employed in them. You will find the name of the author, the method used, the results yielded and the limitations they faced in Table 3.1, 3.2, 3.3.

Table 3.1: The summary of various research that has been done for audio detection in the past from 2020-2024 is presented here.

Author	Method	Results	Limitations
Marium Mateen [1]	Deep learning using RNN and LSTM architecture for urdu deepfake audio detection	The model achieved an accuracy of 91%	The dataset is not big enough and only one model is applied for detection.
Saleem et al. [2]	Utilized Cyclic GANs for Urdu deepfake voice conversion and use Gradient Boosting for classification.	Demonstrated effective voice conversion and detection capabilities with promising but preliminary results.	Dependent on large datasets for effective training. Complexities in Cyclic GAN architectures may limit practical applications.
Valson et al. [3]	Implemented five classical machine learning algorithms including AdaBoost, which utilized features such as speech pause patterns etc	AdaBoost model achieved the highest performance with a 5-fold cross-validation balanced accuracy of 81%	Dataset is limited and features like speech pause patterns take long time for single audio detection
Pham et al. [4]	Assessed multiple deep learning architectures and proposed an ensemble model for enhanced detection.	Achieved best accuracy of 91% on STFT and CQT spectrograms using CNN model	Potential scalability and generalization issues to real-world scenarios were noted as challenges, requiring further validation.
Munir et al. [5]	Developed a novel Urdu deepfake audio dataset using Tacotron and VITS TTS methods for deepfake generation.	The dataset achieved EERs of 0.495 and 0.524 for deepfakes generated by VITS TTS and Tacotron, respectively.	The study's reliance on specific TTS methods might limit its generalization across different spoofing technologies.

Table 3.2: Continued

Author	**Method**	**Results**	**Limitations**
Hamza et al.[6]	Applied multiple machine learning models, including SVM and VGG-16, on the Fake-or-Real dataset	The VGG-16 model shows accuracy of 93%. Lstm shows accuracy of 91 %	Dependence on a specific type of feature extraction and machine learning model may not effective in modern deepfake audios
Zaynab Almutairi and Hebah Elgibreen [7]	Reviewed existing ML and DL methods for detecting audio deepfakes. Discussed different types of attacks and provided a comparative analysis of detection methods.	Identified gaps in current research datasets and methodologies.	Current methods struggle with generalization across diverse scenarios and languages. Further research needed to address these gaps and improve detection effectiveness.
Mcuba et al.[8]	Explored various CNN architectures for deepfake audio detection, utilizing different audio features	The vgg-16 presents best performance with accuracy of 86%	The study's effectiveness is limited by its dataset, which might not represent the full complexity of real-world audio manipulations.
Yi et al. [9]	Surveyed the field of audio deepfake detection, discussing various types of deepfakes, datasets, and detection methods.	Provided a systematic overview and identified gaps in current research, emphasizing the need for large-scale datasets and improved generalization.	Highlighted the challenge of generalizing detection methods to unknown attacks and the need for better interpretability of results.

Table 3.3: Continued

Author	**Method**	**Results**	**Limitations**
Muller et al. [10]	Re-implemented and evaluated twelve audio spoof detection architectures, created a new, realistic dataset of deepfake audios.	Poor generalization of existing models to new, realistic data significant performance drops predicted.	Highlights the challenge of model overfitting to specific benchmarks and the need for models that generalize well across different scenarios.
Govindu et al. [11]	Utilized GANs for generating deepfake audio and XAI techniques for analysis. Implemented FAD for quality assessment of the generated audio.	Demonstrated effective use of XAI tools in interpreting model decisions and provided quantifiable quality measures through FAD	Limited validation on the broader application across different real-world datasets which may affect the generalization of the results.
Mouna Rabhi, Spiridon Bakiras and Roberto Di Pietro [12]	Evaluated the resilience of audio deepfake detectors to adversarial attacks and introduced a novel defense mechanism	Demonstrated that current detectors can be fooled with nearly 100% effectiveness by adversarial examples, significantly reducing detection accuracy	Further testing is required to ensure the effectiveness of the proposed defense across different scenarios and attack types.
Farkhund Iqbal and Abdul rehman[13]	Used advanced feature engineering and machine learning classifiers to detect audio deepfakes on the Fake or Real (FoR) dataset	Achieved an accuracy of 93% on proposed approach	The study's reliance on a specific dataset may affect the applicability of findings to other scenarios or datasets

Table 3.4: Continued

Author	Method	Results	Limitations
Mathew et al.[14]	Developed software for real-time deepfake detection, tested on the ASVspoof 2019 dataset and new real-time data	Showcased effective detection in controlled tests but highlighted challenges in real-time application.	Static model performance degrades in dynamic real-time scenarios, necessitating further model and dataset enhancements.
Jordan J. Bird and Ahmad Lotfi [15]	Use the DEEP-VOICE dataset and machine learning models with hyperparameter optimization.	Achieved avg accuracy of 99% with the Extreme Gradient Boosting model	The study's reliance on the DEEP-VOICE dataset may limit generalization to other audio types or conditions.
Xinfeng et al. [16]	Developed the SafeEar framework which decouples semantic and acoustic information to protect privacy while detecting deepfakes.	Achieving an EER as low as 2.02%.	Challenges in handling varied noisy environments and the complexity of fully ensuring no semantic leakage.
Zhang et al. [17]	Developed RWM for audio deepfake detection that dynamically adjusts learning strategies based on feature distribution characteristics of audio classes.	Achieved an accuracy between 92% to 95.25%	Effectiveness may diminish with less distinctive feature distributions across datasets.
Yuang et al.[18]	Constructed a cross-domain dataset with over 300 hours of speech from zero-shot TTS models; employed diverse attack methods for training	model, resulting in much higher EERs of 29.71% and 44.00%.	The dataset's focus on zero-shot TTS models may not cover all types of deepfake audio technologies.
Anton Firc, Kamil Malinka and Petr Hanacek [19]	Examined various spectrogram types for their utility in deepfake audio detection using deep neural networks.	Demonstrated that different spectrograms have varying efficiencies, with no universally optimal choice	The study's findings are limited by the specific conditions under which the spectrograms were tested, which may not apply universally.
Valente et al. [20]	Utilized CNNs and Mel spectrograms for deepfake audio detection, trained with various datasets	Achieved high accuracy of 96%	Future work needs to focus on adapting to emerging deepfake technologies and improving model adaptability.

3.4 Conclusion

We have read the methodologies and techniques of the different papers that researched on detection of audio deepfakes. Typically used models were classifiers including CNN, SVM, Gradient Boosting and GANS. Different papers used different audio features for detection. Only a few of the research done on urdu language deepfakes.

Chapter 4 Software Requirement Specifications

The following chapter talks about the software requirements, functionalities, and use cases related to the project. It also does a brief risk analysis of the project.

4.1 List of Features

- Ability to upload audio files (in Urdu language) via the web application for classification.
- Feature extraction from recorded audio data
- Detect and classify audios as ”fake” or ”real.”

4.2 Functional Requirements

4.2.1 Audio Upload

- Users must be able to upload audio files via the web application.
- The system should accept multiple file formats, including WAV and MP3.

4.2.2 Converting audio data into digital data

Librosa and MFCC libraries from Python will be utilized to convert the analog audio data into a digital format. This process involves using sampling methods to display continuous analog signals as discrete digital values. These libraries provide tools for signal analysis, feature extraction, and transformation to make audio suitable for machine learning models.

4.2.3 Audio Processing

The MFCC library is used to extract the required audio features from the system. When we run deep learning models , it sets and updates weight to determine which features are important. This is automated. These features are critical inputs for deep learning models, which automatically learn and adjust weights to identify the most relevant audio patterns during training.

4.2.4 Classification of Audio

The system shall process the uploaded audio file and classify it as either ”real” or ”fake”. The system shall optimize classification processes after preprocessing and feature extraction using deep learning models to provide results with minimal delay.

4.2.5 Web Application Interface

The web application shall display a simple user interface where users can upload audio files, view results, and track their classification history. The interface will be designed for ease of use, ensuring accessibility. It will provide clear instructions for uploading files, display results in a user-friendly format, and maintain a history log for users to review past classifications conveniently.

4.3 Quality Attributes

- Accuracy
- Performance
- Security
- Reusability

4.4 Non-Functional Requirements

4.4.1 Security Requirements

The web application shall implement secure login mechanisms for admin access. No personal user data shall be stored unless explicitly authorized by the user. Audio files will be deleted after classification with permission to ensure user privacy. The application will include real-time checks and regular updates to find and fix security issues quickly.

4.4.2 Reusability

The system components, including the classification models, shall be designed with reusability, allowing for easy adaptation to different languages or audio types in the future. The codebase shall be modular, enabling modules (such as the audio processing or classification logic) to be reused in other applications or projects without significant modification.

4.4.3 Performance

The system must be capable of responding to the person instantaneously. The model must process each audio file and return a classification result in under 5 seconds.To achieve this, efficient algorithms and optimized processing techniques will be employed. The system will be tested to make sure it works well even when many users are using it at the same time.

4.5 Assumptions

Following are some assumptions made for the system:

- It is assumed that the dataset in training provided contains a sufficient amount of diverse, high-quality audio samples in Urdu, covering both real and fake audios.
- It is assumed that users will upload audio files in standard formats such as WAV or MP3
- The web application assumes stable internet connectivity for users uploading audio files and receiving results.
- The project assumes that pre-trained models for tasks like text-to-speech or general-purpose audio classification will be available and can be fine-tuned for the specific requirements.
- It is assumed that the majority of audio files uploaded for detection will be of moderate length.
- The system assumes that the audio recordings are made in a relatively quiet and stationary environment, minimizing external factors.

4.5.1 Use Case 1: Example Use Case

4.5.2 Use Case 1: Example Use Case

Name	User recording/uploading audio and getting response use case		
Actors	User		
Summary	This use case involves the process of detecting fake or real audio in recorded audio data.		
Pre-Conditions	The user has access to a device capable of uploading audio.		
Post-Conditions	The system provides the user with the classification result.		
Special Requirements	None		
Basic Flow			
Actor Action		**System Response**	
1	The user initiates the audio detection process by uploading the audio	2	1. The system converts the audio from the individual into a usable format. 2. The system extracts important features from the recorded audio data and classify it as real or fake.
Alternative Flow			
3	The user provides invalid audio data.	4	The system communicates an error message indicating the data is insufficient or corrupted

4.6 Hardware and Software Requirements

4.6.1 Hardware Requirements

- Audio recording/uploading device (e.g. mobile phone, Laptop)
- A computer for processing of data and model development
- Adequate storage space for storing audio datasets, trained models, and project-related files

4.6.2 Software Requirements

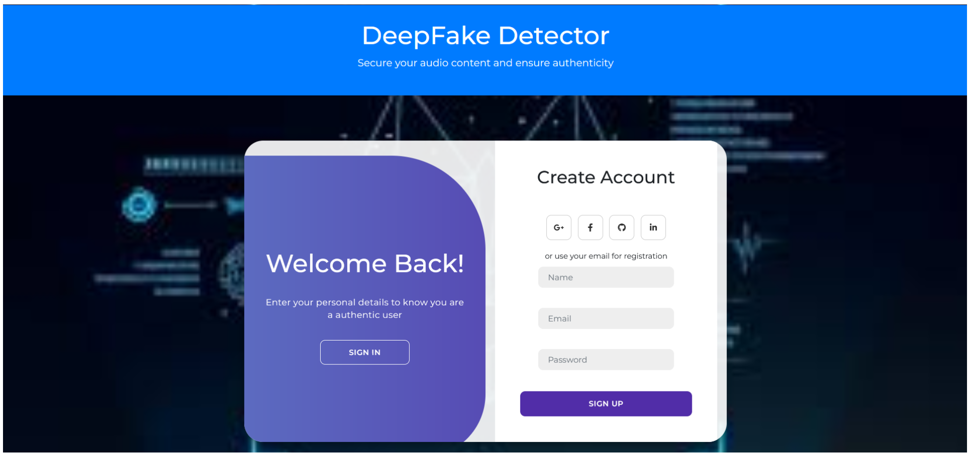
- Visual Studio Code will be used for project development, along with libraries such as PyTorch, TensorFlow, Keras for training and fine-tuning models and Librosa for feature extraction and preprocessing
- Backend Framework: Python flask
- Frontend Framework: HTML, CSS and Bootstrap
- Database: mysql / PHP

4.7 Graphical User Interface

The graphical user interface is shown for different pages along with a description of their working. The users of this website are people who want to verify the authenticity of audio recordings, including everyday users concerned with the rising threat of scams involving fake audio.

4.7.1 Signup Page

The signup page offers an easy registration process for the new users who want to maintain their history. Users are required to fill in basic information, such as their name, email, and password. Once registered, users can access their account, track their classification history, and manage their preferences securely. The process will be simple and quick, ensuring a smooth user experience.



DeepFake Detector

Secure your audio content and ensure authenticity

Welcome Back!

Enter your personal details to know you are a authentic user

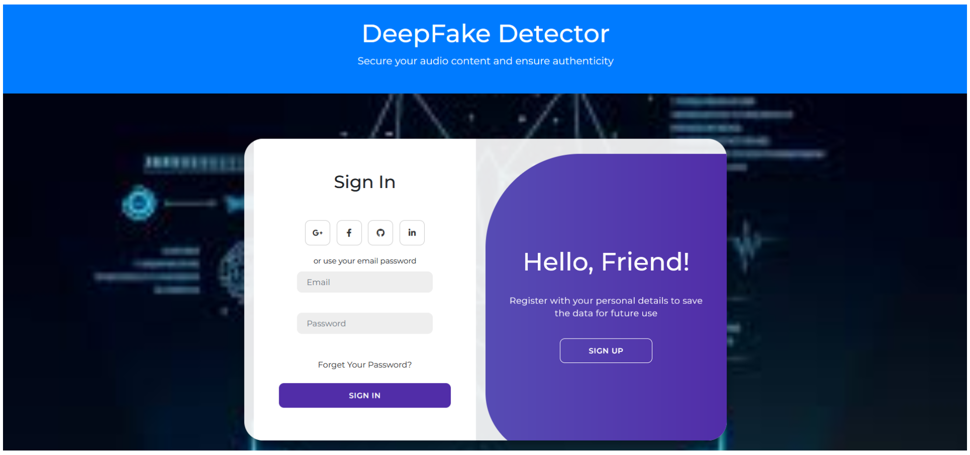
Create Account

or use your email for registration

Figure 4.1: Signup Page for new users

4.7.2 Login Page

The login page provides users with a simple and straightforward way to access their accounts by entering their email and password.



DeepFake Detector

Secure your audio content and ensure authenticity

Sign In

or use your email password

Hello, Friend!

Register with your personal details to save the data for future use

Figure 4.2: The login interface for users

4.7.3 Home Page

On this page, users can upload an audio file to be analyzed for authenticity. After processing, the system displays the results, identifying whether the audio is real or fake.

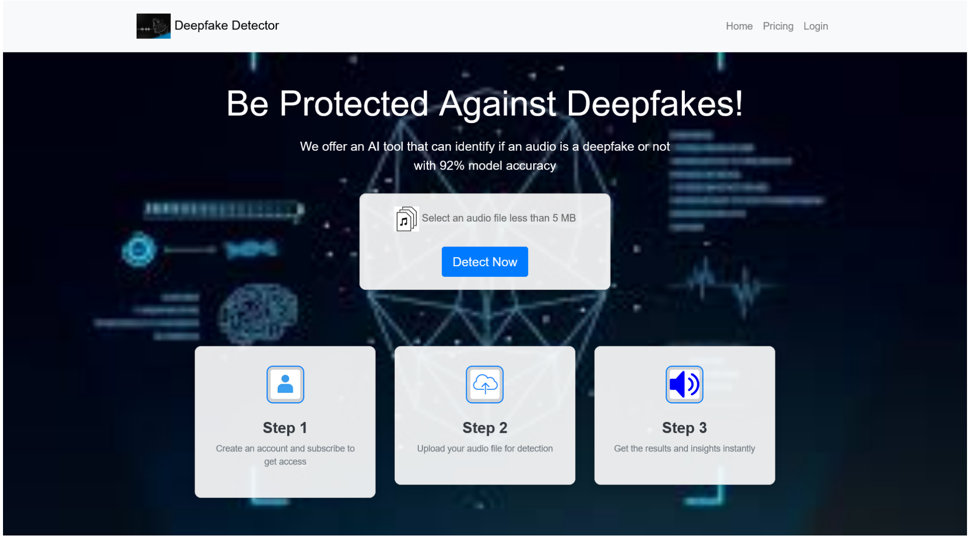

Deepfake Detector
Home Pricing Login
Be Protected Against Deepfakes!
We offer an AI tool that can identify if an audio is a deepfake or not with 92% model accuracy
Select an audio file less than 5 MB
Detect Now
Step 1
Create an account and subscribe to get access
Step 2
Upload your audio file for detection
Step 3
Get the results and insights instantly

Figure 4.3: Home page of the deepfake audio detection

4.8 Risk Analysis

- Insufficient or inaccurate audio training data can result in a model that fails to accurately detect real or fake audios.
- The model could become biased if the dataset is not diverse enough, which could lead to inaccurate predictions.
- Pre-trained models may not adapt well to the Urdu language, requiring extensive fine-tuning and increasing the risk of poor accuracy.

4.9 Conclusion

This chapter discusses the functional and non-functional requirements, along with other software and hardware prerequisites necessary for the development of our system. Furthermore, it provides an in-depth analysis of use cases, discussing their primary and alternative flows. Additionally, potential risks associated with this project are also examined.

Chapter 5 Proposed Approach and Methodology

Our deepfake audio detection project focuses on classifying both AI-generated and human-generated audio samples, specifically tailored to meet the needs of Urdu language processing. Given that Urdu datasets are limited, We used LUMS- generated dataset, creating a comprehensive resource. This chapter outlines our methodology, which includes cleaning, feature extraction, labeling, and dataset splitting, leveraging deep learning and audio processing techniques to reliably distinguish between real and synthetic audio.

5.1 Audio Dataset

In this phase, we will work with a premade dataset developed by LUMS University, which consists of Urdu news-related dialogues recorded under controlled conditions. The recordings, saved in high-quality audio formats (e.g., WAV), will facilitate a thorough analysis. The dataset distinguishes between audio generated by AI (labeled as spoofed) and that produced by human speakers (labeled as bonafide). Our bonafide dataset includes samples from 17 speakers (7 females and 10 males), ensuring a representative mix of voices. This pre-collected dataset allows us to focus on analyzing and detecting deepfakes efficiently.

5.2 Audio Cleaning and Feature Extraction

The dataset will undergo a cleaning process to enhance its quality. This phase will address noise reduction and inconsistencies, employing libraries such as Librosa, PyDub. Since the audio recordings may vary in length, we will apply zero padding to standardize their durations, facilitating easier implementation during the testing phase. Following the cleaning process, we will extract key features, specifically MFCCs, Short time Fourier Transform (STFT), Chroma features calculated with Librosa—are extracted to obtain comprehensive spectral information regarding pitch, tone, and rhythm. These features will be organized alongside participant metadata into a comprehensive dataframe, preparing it for subsequent analysis.

5.3 Data Labeling

Once the features are extracted, we will label the dataset to support supervised learning. Each audio instance will be classified as either bona fide (1) or spoofed (0). This binary classification is essential for effective model training and evaluation. The clear distinction between real and fake audios will enable the model to learn patterns that help in reliable detection of deepfake audio, ensuring accuracy and efficiency in real-world scenarios. This approach will provide a more nuanced understanding of the data, aiding in the development of robust detection models.

5.4 Data Splitting and Normalization

To ensure the model's generalizability, we will split the dataset into training, validation, and test sets while maintaining a balanced representation of bona fide and spoofed audio. This stratification will help minimize bias and ensure diversity in training examples. Normalization of the MFCC features will further reduce variances in recordings, enhancing the model's resilience and ability to generalize effectively across different audio samples.

5.5 Model Development

In the model development phase, we will explore a range of machine learning and deep learning techniques specifically designed for audio analysis. The labeled dataset will serve as the foundation for train- ing models capable of distinguishing between bonafide and spoofed audio. We will evaluate various machine learning algorithms, including Support Vector Machines (SVM), and logistic regression to capture complex audio patterns. Additionally, we will implement deep learning architectures such as ResNet, transformer models(Pre-trained transformer models like Wav2Vec 2.0 or HuBERT) and Convolutional Neural Networks (CNN), which are particularly well-suited for sequential data analysis. Data augmentation techniques will be employed to improve the model's adaptability to real-world scenarios. Throughout the training process, we will track performance metrics, including accuracy, precision, recall, and F1 score, to assess and refine our models.

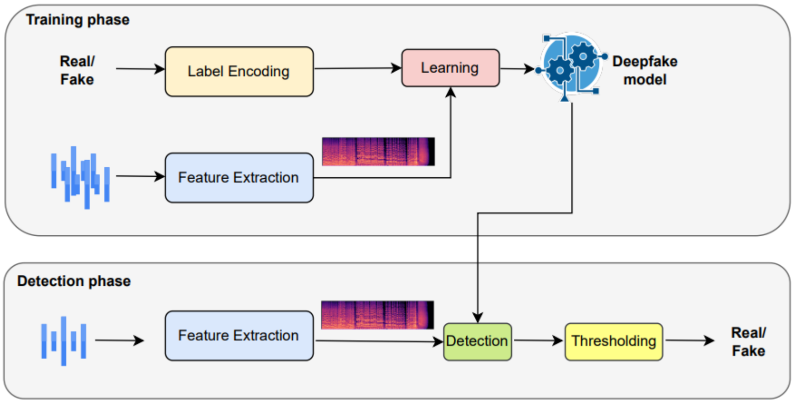


Figure 5.1: Pipeline of deepfake audio detection system

5.6 Conclusion

This chapter presents a detailed methodology for our deepfake audio detection project, outlining the systematic approach from audio data collection to model development. By focusing on both AI-generated and human-generated audio, we aim to create a reliable detection system specifically for the Urdu language. The comprehensive data cleaning, feature extraction, labeling, and normalization processes will lay a solid foundation for our models. By utilizing a combination of machine learning and deep learning techniques, we intend to contribute to addressing the challenges posed by deepfakes in low-resource languages, with performance metrics guiding our continuous improvement efforts.

Chapter 6 High-Level and Low-Level Design

6.1 System Overview

In this section, we will discuss the overall overview of the Deepfake Audio detection system.

6.1.1 Audio Data Set

The system accepts audio recordings from users through a web-based application which allows them to upload clips. These clips then further are analyzed and results are shown to user. The dataset primarily consists of Urdu news audios, which is used for training and testing various deep learning models. The system's user interface ensures an intuitive and easy experience for users to provide audio data for deepfake detection.

6.1.2 Preprocessing and Feature Extraction

Preprocessing is performed to enhance its quality. This involves noise reduction, zero padding for length standardization, also to digitize analog audio data and handling inconsistencies using libraries such as Librosa and PyDub. Noise reduction will reduce the background noises. Following preprocessing, key features such as Mel Frequency Cepstral Coefficients (MFCCs), Short-Time Fourier Transform (STFT), and Mel Spectrograms are extracted. They capture essential details about pitch, tone, and rhythm. These features serve as the input for the classification models.

6.1.3 Deepfake Detection

After feature extraction, the system applies machine learning and deep learning models, including Support Vector Machines (SVM), logistic regression, and deep learning architectures like ResNet, transformer models (e.g., Wav2Vec 2.0 or HuBERT), and CNN. These models classify audio as either real (bona fide) or fake (spoofed). The model selection is based on their ability to capture complex audio patterns effectively and provide high accuracy.

6.1.4 Design Approach

The system follows a modular design approach to ensure scalability, flexibility, and easy integration of new models or features in the future. Each separate module, promotes reusability and simplified updates. The web app interface provides a seamless user experience, enabling users to upload audio files, view classification results, and interact with the system in real-time.

6.1.5 Organization

The system's functionality is organized into several key stages. Each of these stages plays a critical role in delivering the final classification results to the user through the web interface.These stages include audio file upload, preprocessing, feature extraction, classification, and result presentation. Each step is carefully designed to ensure seamless processing and efficient delivery of accurate results to the user.

6.1.6 Risk Analysis

Key risks include the challenge of acquiring diverse training data for real-time applications and ensuring smooth real-time classification through the web app interface. Additionally, varying internet connectivity and server load could affect the system's performance, but these are mitigated by robust backend infrastructure and efficient algorithms.

6.2 Design Considerations

This section describes many of the issues that need to be addressed or resolved before attempting to devise a complete design solution.

6.2.1 Assumptions and Dependencies

Following are the issues that concern the following areas:

6.2.1.1 Hardware Device

The system assumes that users have access to devices with the ability to upload audio files, as well as an internet connection to interact with the web interface. If any failure interruption occurs, it can be a challenge in the classification process.

6.2.1.2 End-User Characteristics

The system assumes that the users are non-technical people who want to verify the accuracy of the recordings. So the interface is designed to be simple and intuitive. Any inconsistency from genuine audio characteristics may compromise the quality of the dataset, which in turn can impact the model's accuracy and reliability in detecting deepfake audio.

6.2.1.3 Probable Changes in Functionality

As deepfake technologies evolve, new challenges may emerge.The system is designed with flexibility in mind, enabling updates to integrate detection patterns or other methods without significant disruption.The ability of the system to accommodate changes in usage depends on being well- documented code and modular design.

6.2.2 General Constraints

- Variations in the audio recording environment can introduce background noise or distortions, affecting the quality of audio data. The system must account for these environmental factors to ensure accurate detection of deepfake audio.
- Limited computing resources may impact the system's performance, particularly during real-time audio processing. The design must consider these resource limitations to maintain efficiency and responsiveness.
- The audio data and extracted features are stored securely in a database for further analysis. This repository needs to be both secure and efficient to allow quick access to data while adhering to data privacy regulations.
- Comprehensive testing is essential to meet validation and verification requirements. Inadequate testing can lead to undetected errors, diminishing the accuracy and reliability of deepfake audio detection.

6.2.3 Goals and Guidelines

- Adhering to the KISS principle (”Keep it simple, stupid!”) is a primary goal. It emphasizes simplicity in design, easier for users to navigate and enhance their experiences. Simple systems are also easier to maintain.
- Accuracy is an important part of this system, as higher accuracy of the system leads to better identification of authentic audio.
- The system's design must prioritize scalability to accommodate potential enhancements in functionality or increases in data volume.

6.2.4 Development Methods

Agile methodology, particularly the Scrum framework, will be our development process. Scrum is a flexible and straightforward framework that helps manage complex projects effectively. The work is divided into fixed-length iterations known as sprints, allowing for regular assessment and adaptation to ensure valuable outcomes. Daily stand-up meetings are held to discuss the tasks planned for the day, while sprint review meetings take place at the end of each sprint to evaluate progress and gather feedback. Additionally, sprint retrospective meetings are scheduled before the next sprint begins to identify objectives and any necessary adjustments, focusing on continuous improvement throughout the project.

6.3 System Architecture

The diagram below breaks down our system into several components like Model Development and preprocessing. We also have a pre-processed dataset in which our dataset is stored. This data will be used to train the model. The raw input will be uploaded by the user and then the raw data will be pre-processed and given to the model development component. After that, the detection result will be displayed on the console.

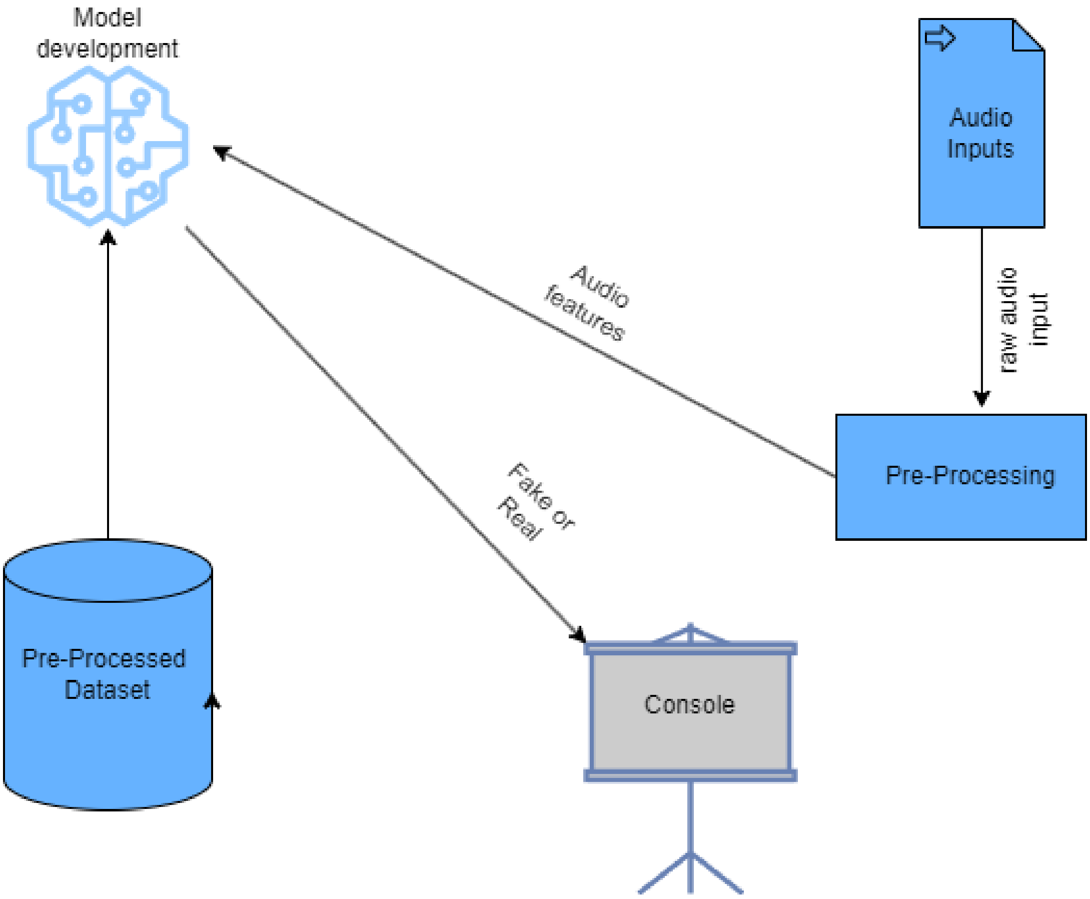


Figure 6.1: High level system architecture design

6.3.1 Pre-Processing

The figure above explains the steps of pre-processing after the audio is collected its cleaning is done and the features are extracted through the libraries of Python mentioned previously.

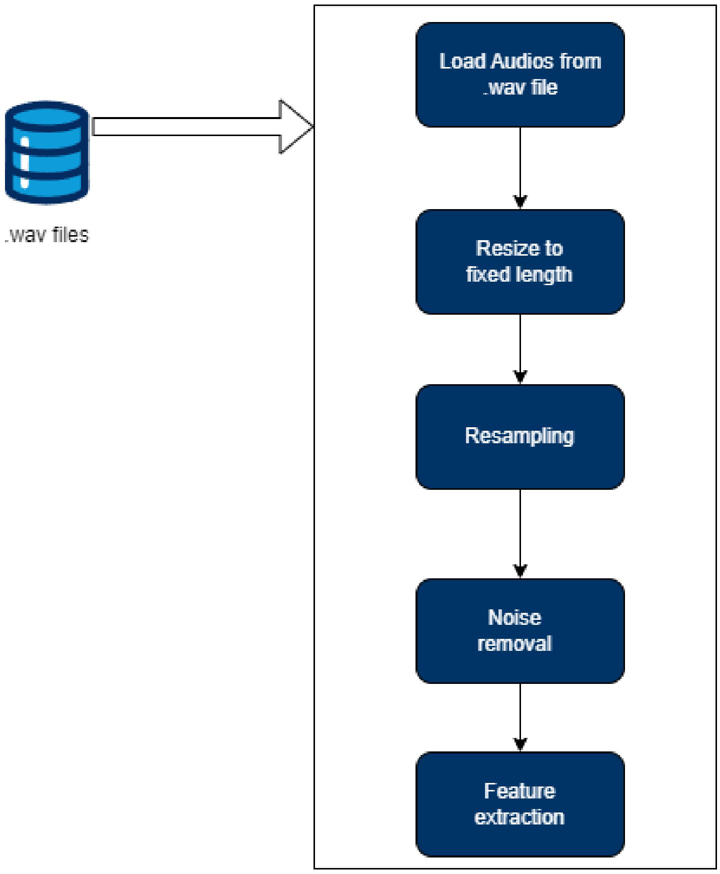


Figure 6.2: Low level architecture design for Pre-processing

6.3.2 Model Development

The diagram below explains the model development component. We have used two main models which further include CNN, vision transformer in the Deep learning model and SVM in the Machine Learning Model. These two models combine to make the overall model development component

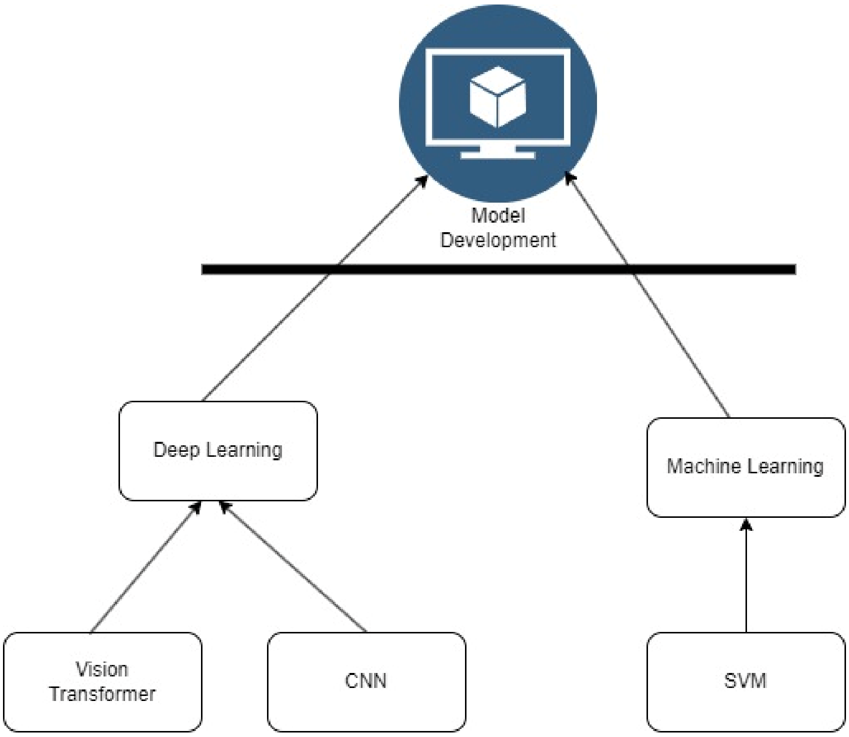


Figure 6.3: Low level architecture design

6.3.3 Dataset

The deepfake audio data contains audio recordings in urdu which are classified as bonafide and AI generated. The dataset is recorded by 17 speakers including 10 male and 7 females. The original audio by these 17 speakers are then converted into AI generated using TTS models. In this way each speaker has the original audio and 2 kinds of AI generated audios.

6.4 Architectural Strategies

6.4.1 Utilization of a Deep Learning Model

To achieve accurate deepfake detection from audio data, the architectural strategy involves employing a deep learning model, particularly CNN and vision transformer. The decision to use a CNN is due to its effectiveness in capturing sequential patterns and features within audio data. Vision transformers are the new advancement that will take images as input and predict the class. Machine learning models include SVM, but these models are not preferred because of their limited capabilities that are not for complex tasks. Furthermore, most of the models used in the literature review phase were CNN or variations of it. This highlights the popularity and robustness of the model.

6.4.2 Cloud based storage

To store the user's audio data for deepfake detection in a scalable and secure manner, cloud-based storage solutions such as Google Cloud Storage were used to ensure secure, easily accessible storage and the data is repaired in case the user has visited the website, which is a loss at the time of transfer. When the user uploads a .wav audio file to the website, the audio will be compressed, then sent directly to the cloud using the HTTPS protocol and rest.Meta data such as , user id , upload date and the results... database. This approach will reduce security risks and backup and retention policies will keep the data consistent.

6.4.3 Modular Model pipeline

Adopting a modular model pipeline was a key architectural strategy that ensured maintainability and reusability. The codebase is organized into distinct modules, each responsible for specific functionalities such as audio data processing, feature extraction, and model training. This design decision allowed for easy updates and future enhancements. Alternative strategies, like a monolithic codebase, were evaluated but rejected in favor of modularity to facilitate efficient code management

6.4.4 Transfer Learning

In transfer learning we will fine-tune a pre-trained models used in audio deepfake detection applied for some other languages like Chinese, English etc. This will allow us to see the difference in model accuracies if we are doing it from scratch and fine tuning pre-training models. This also helps in reducing training time as well.

6.5 Domain Model/Class Diagram

The system follows a comprehensive audio processing pipeline to identify deepfakes from recorded data. Beginning with the raw audio signal, the process initiates with audio pre-processing, involving tasks like noise reduction, audio resampling and duration limitation. Subsequently, the feature extraction module employs MFCCs, Mel-Spectrograms and other features to capture fake and real audio patterns. During prediction, the model evaluates the extracted features, employing learned weights to classify fake and real probabilities.

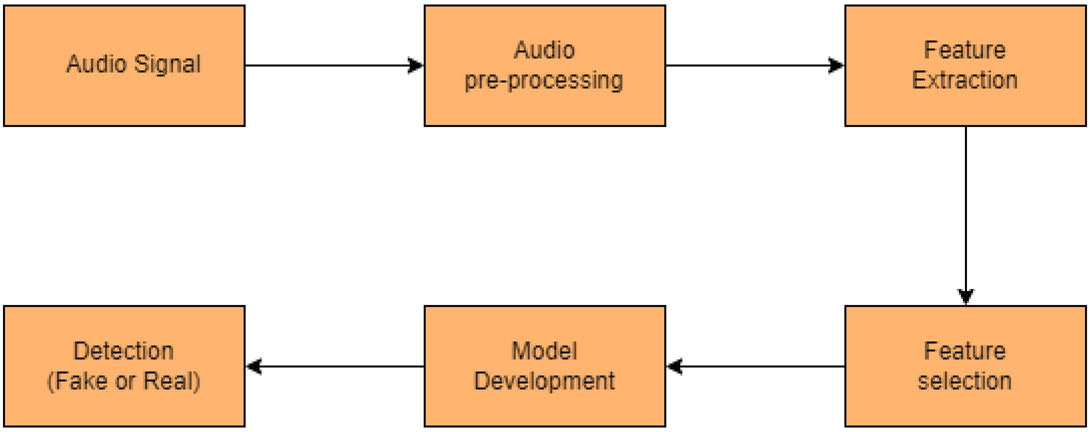


Figure 6.4: Model Diagram of low level architecture

6.6 Policies and Tactics

6.6.1 Coding Guidelines and Conventions

The project follows a set of conventions and guidelines for consistency and maintainability in coding. These include a well-defined and documented structure for the source code organization, as well as standard naming conventions for variables, functions, and classes. Additionally, the project's programming phase would adhere to the standards and guidelines established in Python. To maintain the readability of our code, we will concentrate on styling guidelines

6.6.2 Testing Strategy

A comprehensive testing strategy will be employed to ensure the reliability of the system. Unit testing will be implemented, allowing us to verify individual components. Integration testing will follow, focusing on the interactions between various modules. Validation and Testing of the machine learning modules used will be done across various metrics like F1 score, precision accuracy and recall.

6.6.3 Maintenance of Software

The software will be kept updated and further functionalities can be added such as enhancing the dataset. We will ensure that everything is kept up to date including the models and the datasets with appropriate labels so that it is efficient to use with the models. Regular maintenance will also include fixing bugs, improving performance, and incorporating user feedback to ensure the system continues to meet evolving needs.

6.7 Conclusion

In conclusion, this chapter has provided a comprehensive overview of our project, delving into both its high-level and low-level designs. The interactions between the various system components were demonstrated through detailed component and architecture analysis. Collectively, these elements lay the groundwork for the development and implementation of our project.

Chapter 7 Implementation and Test Cases

This chapter outlines the implementation of our project, focusing on detecting deepfake audio in the Urdu language. It encompasses dataset preparation, feature extraction, model development, and evaluation, systematically explaining the steps undertaken to achieve accurate results.

7.1 Implementation

Our project aimed to develop a binary classification system to detect bonafide and fake audio (deepfakes) using a structured approach. This section describes the steps from dataset preparation to implementing machine learning and deep learning models.

7.1.1 Implementation of Input and Data Cleaning

The Urdu Audio Deepfake Detection dataset contains recordings from 17 speakers, divided into four categories: Bonafide Part 1 (12,036 files), Bonafide Part 2 (8,415 files), Spoofed Tacotron (8,415 files), and Spoofed VITS TTS (8,415 files). Each category is stored in a well-structured folder hierarchy, ensuring easy accessibility for processing. Preprocessing was a key step in ensuring the quality and uniformity of the audio data. The audio files were trimmed to remove unnecessary silence, resampled to 16kHz for standardization, and processed for noise reduction to eliminate background artifacts. These preprocessing steps enhanced the clarity of the audio data, making it suitable for feature extraction and subsequent model training. We tackled noise reduction and inconsistency resolution using libraries like Librosa, PyDub, pandas, NumPy, and SciPy.

7.1.2 Implementation of Feature Extraction

The feature extraction phase of the Deepfake audio detection project was methodically carried out with Pandas and Librosa libraries. WAV files were loaded and processed for features like MFCCs, Mel spectrograms, Log-mel spectrograms, Pitch, Frequency and Zero crossing Rate. MFCCs were used to capture detailed and compact information about the sound, such as its pronunciation and tone, we can analyze its spectral characteristics. Additionally, raw waveforms retained the original audio signal for models capable of end-to-end learning. This diverse set of features ensured that the models were exposed to a wide range of audio characteristics, enhancing their ability to differentiate between bonafide and spoofed audio.

7.1.3 Implementation of Data Labeling

The dataset was labeled to reflect the binary classification task. Bonafide audio files were labeled as 0, while the spoofed audio files from the Tacotron and VITS TTS categories were labeled as 1. This labeling provided the necessary ground truth for supervised learning. Accurate and consistent labeling was critical to ensure the models could effectively learn the distinction between real and fake audio.

7.1.3.1 Data Splitting

To ensure a robust evaluation, the dataset was split into training and testing sets, with 80% of the data used for training and 20% reserved for testing. Stratified sampling was employed to maintain the proportional distribution of bonafide and spoofed files in both subsets. This approach minimized the risk of data imbalance, which could otherwise lead to biased model performance.

7.1.4 Implementation of Model Development

The following models were implemented and tested using different input features derived from the dataset:

7.1.4.1 Support Vector Machine (SVM)

SVM, a traditional machine learning algorithm, was explored to classify deepfake audios. Two feature types were used as input : MFCCs (Mel Frequency Cepstral Coefficients), which represent spectral characteristics of audio, and HuBERT embeddings, which provide a deep semantic representation of audio signals. SVM with MFCC features achieved moderate accuracy, highlighting the limited discriminative capacity of shallow features. However, using HuBERT embeddings significantly improved the model's accuracy, demonstrating the power of deep audio representations.

7.1.4.2 Decision Tree

The Decision Tree model was implemented to explore rule-based learning for audio classification. This model works by recursively splitting the feature space based on the most informative features at each node, resulting in a tree-like structure. The input features used were MFCCs, derived from audio data. While the decision tree model was simple and interpretable, it underperformed compared to other approaches. Its tendency to overfit to small feature variations in the dataset limited its effectiveness in generalizing to unseen data.

7.1.4.3 Convolutional Neural Network (CNN)

CNN was used to exploit spatial patterns in audio features. The architecture consisted of input layers for feeding Mel spectrograms, MFCCs, and full spectrograms, convolutional layers for extracting feature maps, pooling layers for dimensionality reduction, and fully connected layers for classification. Mel spectrograms, which represent the power spectrum of audio, provided the most informative input features, outperforming MFCCs and raw spectrograms. The CNN leveraged the rich frequency-time representation of Mel spectrograms, yielding superior performance compared to other features.

7.1.4.4 Vision Transformer (ViT)

The Vision Transformer (ViT) was applied to capture global dependencies in audio feature representations. The architecture divides input features into smaller patches and processes them through a transformer encoder equipped with attention mechanisms, concluding with a classification head. Mel spectrograms, treated as 2D image-like inputs, were primarily used, with MFCCs as an alternative input representation. The ViT model excelled with Mel spectrograms, effectively leveraging attention mechanisms to understand audio characteristics and achieving high accuracy.

7.1.4.5 Neural Network with HuBERT Embeddings

A neural network architecture was designed to utilize HuBERT embeddings as input for audio classification. The model featured an input layer for HuBERT embeddings, followed by fully connected layers that captured feature dependencies and processed them through non-linear activation functions. The final classification head consisted of a fully connected layer for binary classification. By leveraging the rich representation of HuBERT embeddings, this neural network achieved superior performance compared to other models. It highlighted the effectiveness of combining deep audio representations with neural network architectures for accurate and efficient audio classification.

7.1.5 Model Evaluation

After completing the development of various models, we evaluated them on the test set using critical metrics to assess their effectiveness. Metrics such as accuracy and loss provided an overall indication of model performance, while metrics like precision, recall, and F1 score offered deeper insights into the models' ability to differentiate between real and spoofed audio cases. By analyzing these metrics, we identified the strengths and limitations of each approach, ensuring a comprehensive understanding of their capability to classify the audio data accurately.

7.2 Test case Design and description

All the test cases performed during the testing phase of the system are elaborated below.

Table 7.1: Audio file Test case No.1

Audio File Upload Test Case			
Test Case ID:	*1*	**QA Test Engineer:**	*Fayez*
Test case Version:	*1*	**Reviewed By:**	*Zahra*
Test Date:	*08-04-2025*		
Revision History:	*none*		
Objective:	*Verify that the audio file selected is of correct type*		
Product/Ver/ Module:	*Audio File Upload module*		
Environment:	*Windows operating system installed.*		
Assumptions:	*Assumes that user has audio files to upload.*		
Pre-Requisite:	*none*		
Step No.	**Execution description**	**Procedure result**	
1	*Clicks browse audio file button.*	*Window pops up showing user the local memory.*	
2	*Selects wrong file..*	*Error window pops up.*	
Comments: The test case is passed. Our system is working properly.			
Passed			

Table 7.2: Deepfake Detection Test case No.2

Deepfake detection Test Case			
Test Case ID:	*2*	**QA Test Engineer:**	*Roohan*
Test case Version:	*2*	**Reviewed By:**	*Fayez*
Test Date:	*08-04-2025*		
Revision History:	*none*		
Objective:	*Verify that the result of selected audio file is correct.*		
Product/Ver/ Module:	*Audio deepfake detection module*		
Environment:	*Windows operating system installed.*		
Assumptions:	*Assumes that user has the audio files*		
Pre-Requisite:	*none*		
Step No.	**Execution description**	**Procedure result**	
1	*Clicks detect button.*	*Window pops up showing the results.*	
Comments: The test case is passed. Our system is working properly.			
Passed			

Table 7.3: Sample Test case Matric.No.1

Metric	Purpose
Number of Test Cases	2
Number of Test Cases Passed	2
Number of Test Cases Failed	0
Test Case Defect Density	0
Test Case Effectiveness	0

7.3 Conclusion

We evaluated different models and features for detecting deepfake audio. Each model was tested thoroughly using key metrics like accuracy, precision, recall, and F1 score to determine their strengths and weaknesses. The results showed that models using deep features, such as HuBERT embeddings and advanced architectures, such as Vision Transformers, performed significantly better compared to traditional models with simpler features. These findings highlight the importance of combining deep feature extraction with modern architectures to tackle complex audio classification tasks.

Chapter 8 Experimental Results and Discussion

This chapter presents experimental results conducted on the prototype we have developed so far. The evaluation metrics, including accuracy, precision, recall, and F1 score, provide insights into the effectiveness of the models and features used.

8.1 Comparative Analysis of Model Performance

To evaluate the effectiveness of the various models used for deepfake audio detection, a comparative analysis was conducted based on key evaluation metrics, including accuracy, precision, recall, and F1 score. The table below highlights the performance of each model across different feature sets, showcasing the impact of both feature representation and model architecture. This comparison highlights the progress from traditional machine learning models to advanced deep learning approaches.

Table 8.1: Evaluation Metrics for Deepfake Audio Detection Models

Model	**Features**	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Decision Tree	MFCCs	78.5%	74.2%	76.3%	75.2%
SVM	HuBERT Embeddings	85.6%	82.5%	84.8%	83.6%
SVM	MFCCs	70.2%	68.5%	69.8%	69.1%
CNN	Mel Spectrograms	88.7%	87.2%	89.1%	88.1%
CNN	MFCCs	86.1%	84.7%	85.9%	85.3%
Vision Transformer	Mel Spectrograms	90.5%	89.8%	91.2%	90.5%
Vision Transformer	MFCCs	87.6%	86.3%	87.9%	87.1%
Neural Network	HuBERT Embeddings	**92.4%**	**91.5%**	**92.9%**	**92.2%**

8.2 Conclusion

The experimental analysis showed that the improved models with deep feature representations that outperform traditional methods in detecting deepfake audio.This information opens the way For a sophisticated and reliable solution in deep fake detection.

Chapter 9 Conclusions

The primary aim of this project was to detect Urdu audio deepfakes using advanced machine learning and deep learning techniques. The focus on Urdu, a low-resource language, adds significant value to this work, as it addresses a critical gap in existing deepfake detection research. Through rigorous dataset preparation, exploration, and experimentation with various models, substantial progress was achieved during FYP-1.

In FYP-1, key milestones included preprocessing and analyzing the provided Urdu audio dataset to ensure its readiness for training. Various classification models, including traditional machine learning methods like SVM and Decision Trees, and deep learning approaches like CNNs, Vision Transformers, and custom architectures, were implemented and evaluated. HuBERT embeddings and MFCCs were explored as feature representations, highlighting their strengths in improving model performance. The foundational work laid during FYP-1 has created a solid platform for further refinements.

In FYP-2, significant progress has been made in both model optimization and system deployment. The custom model architecture have been fine-tuned to enhance accuracy and performance, ensuring more reliable deepfake detection. Additionally, a web application has been developed using Python Flask, integrating both frontend and backend components to allow real-time deepfake audio detection. This platform provides users with a seamless experience to upload audio files and receive classification results instantly. While the core functionality has been implemented, the website is still under development, and additional features will be added to enhance its usability and effectiveness.

In conclusion, the project has laid a strong foundation for detecting Urdu audio deepfakes by addressing preprocessing, model exploration, and initial evaluation. The combination of optimized models and a web-based interface brings practical usability to deepfake detection in the Urdu language, addressing real-world challenges. Ongoing enhancements to the web application will make it even more robust and user-friendly. This work represents a meaningful contribution to the field and provides a pathway for continued advancements in combating audio deepfake challenges.

Bibliography

[1] M. Mateen, "Deep learning approach for detecting audio deepfakes in urdu," *Journal/Conference Name (NUML)*, 2023.

[2] S. Saleem, A. Dilawari, M. U. G. Khan, and M. Husnain, "Voice conversion and spoofed voice detection from parallel english and urdu corpus using cyclic gans," in *2019 International Conference on Robotics and Automation in Industry (ICRAI)*, pp. 1–6, IEEE, 2019.

[3] N. V. Kulangareth, J. Kaufman, J. Oreskovic, and Y. Fossat, "Investigation of deepfake voice detection using speech pause patterns: Algorithm development and validation," *JMIR biomedical engineering*, vol. 9, p. e56245, 2024.

[4] L. Pham, P. Lam, T. Nguyen, H. Nguyen, and A. Schindler, "Deepfake audio detection using spectrogram-based feature and ensemble of deep learning models," *arXiv preprint arXiv:2407.01777*, 2024.

[5] S. Munir, W. Sajjad, M. Raza, E. Abbas, A. H. Azeemi, I. A. Qazi, and A. A. Raza, "Deepfake defense: Constructing and evaluating a specialized urdu deepfake audio dataset," in *Findings of the Association for Computational Linguistics ACL 2024*, pp. 14470–14480, 2024.

[6] A. Hamza, A. R. R. Javed, F. Iqbal, N. Kryvinska, A. S. Almadhor, Z. Jalil, and R. Borghol, "Deepfake audio detection via mfcc features using machine learning," *IEEE Access*, vol. 10, pp. 134018–134028, 2022.

[7] Z. Almutairi and H. Elgibreen, "A review of modern audio deepfake detection methods: challenges and future directions," *Algorithms*, vol. 15, no. 5, p. 155, 2022.

[8] M. Mcuba, A. Singh, R. A. Ikuesan, and H. Venter, "The effect of deep learning methods on deepfake audio detection for digital investigation," *Procedia Computer Science*, vol. 219, pp. 211–219, 2023.

[9] J. Yi, C. Wang, J. Tao, X. Zhang, C. Y. Zhang, and Y. Zhao, "Audio deepfake detection: A survey," *arXiv preprint arXiv:2308.14970*, 2023.

[10] N. M. Müller, P. Czempin, F. Dieckmann, A. Froghyar, and K. Böttinger, "Does audio deepfake detection generalize?," *arXiv preprint arXiv:2203.16263*, 2022.

[11] A. Govindu, P. Kale, A. Hullur, A. Gurav, and P. Godse, "Deepfake audio detection and justification with explainable artificial intelligence (xai)," 2023.

[12] M. Rabhi, S. Bakiras, and R. Di Pietro, "Audio-deepfake detection: Adversarial attacks and countermeasures," *Expert Systems with Applications*, vol. 250, p. 123941, 2024.

[13] F. Iqbal, A. Abbasi, A. R. Javed, Z. Jalil, and J. N. Al-Karaki, "Deepfake audio detection via feature engineering and machine learning.," in *CIKM Workshops*, 2022.

[14] J. J. Mathew, R. Ahsan, S. Furukawa, J. G. K. Kumar, H. Pallan, A. S. Padda, S. Adamski, M. Reddiboina, and A. Pankajakshan, "Towards the development of a real-time deepfake audio detection system in communication platforms," *arXiv preprint arXiv:2403.11778*, 2024.

[15] J. J. Bird and A. Lotfi, "Real-time detection of ai-generated speech for deepfake voice conversion," *arXiv preprint arXiv:2308.12734*, 2023.

[16] X. Li, K. Li, Y. Zheng, C. Yan, X. Ji, and W. Xu, "Safeear: Content privacy-preserving audio deepfake detection," *arXiv preprint arXiv:2409.09272*, 2024.

[17] X. Zhang, J. Yi, C. Wang, C. Y. Zhang, S. Zeng, and J. Tao, "What to remember: Self-adaptive continual learning for audio deepfake detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 19569–19577, 2024.

[18] Y. Li, M. Zhang, M. Ren, M. Ma, D. Wei, and H. Yang, "Cross-domain audio deepfake detection: Dataset and analysis," *arXiv preprint arXiv:2404.04904*, 2024.

[19] A. Firc, K. Malinka, and P. Hanáček, "Deepfake speech detection: A spectrogram analysis," in *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, pp. 1312–1320, 2024.

[20] L. P. Valente, M. M. de Souza, and A. M. Da Rocha, "Speech audio deepfake detection via convolutional neural networks," in *2024 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pp. 1–6, IEEE, 2024.