Implementation of Bengali Voice Signature Authentication Techniques in Financial Systems Using Deep Learning

by

Mahinur Rashid
22301714
Safwan Ahmad
21101311
Faiza Binte Arif
21201498
Ramisa Anjum
21301399
Sanjida Arefin Rimu
21201655

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
October 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Mahinur Rashid

22301714

Safwan Ahmad

21101311

Faiza Binte Arif

21201498

Ramisa Anjum

21301399

Sanjida Arefin Rimu

21201655

Approval

The thesis/project titled "Implementation of Voice Signature Authentication Techniques in Financial Systems Using Deep Learning" submitted by

1. Mahinur Rashid (22301714)
2. Safwan Ahmad (21101311)
3. Faiza Binte Arif (21201498)
4. Ramisa Anjum (21301399)
5. Sanjida Arefin Rimu (21201655)

Of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 10, 2025.

Examining Committee:

Supervisor:
(Member)

Dr. Farig Yousuf Sadeque
Associate Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

In developing-nations like Bangladesh, the need for more secure user-friendly authentication methods to protect financial transactions are a substantial need nowadays due to the growing digitalization in finances. Nowadays the need for more secure, fast and user-friendly authentication methods in the financial sectors is increasing rapidly and here the voice-based authentication provides an aspiring substitution to the regular conventional methods. The aim of this paper is to discuss the area of coverage for the potential application of deep learning models in reference to implementing voice signature authentication procedures for cash-out processes in physical banking and digital mobile payment systems. The methodology includes the development and testing of a voice verification system and its integration with the already existing banking and mobile payment infrastructures. Initially, the Bengali voice data samples are collected from available sources, and then analyzed using statistical methods based on differentiable voice/tone metrics and thematic analysis for purposes of user identification and detection of fraud and AI generated voices. It greatly relies on the use of deep learning techniques in voice signature analysis and authentication along with ensuring the authentication procedures are safe and smooth. Key findings support this study in regarding voice signature authentication process, as it can be a promising and cheap addition to the security protocols by increasing accuracy, reliability and also reduce fraud along with improving smooth user experience in cash-out processes. Additionally, this study offers real-world proof of the successful use of deep learning in financial authentication systems, this study adds up the study to the vast domain of deep learning. This study intends to contribute to the improvement of financial security methods through preserving the sensitive financial data and by tackling the growing need for more secure and user-friendly intuitive authentication techniques. Subsequent future research could dive deeper into voice verification technology and its application in other sensitive sectors for further developments; opening a new door of authentication protocol in a country like Bangladesh.

The entire thesis repository is available here: https://github.com/mahinur-rashid/THESIS-Repository

Keywords: Bengali voice authentication, deep learning, financial security, voice signature, fraud detection, mobile payment systems, user-friendly authentication.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the modern world with the rapid advancement in technology and increasing dependency on digital transactions, the need for a near-perfect security system is a growing fundamental need. Some of the traditional authentication methods like signature or passwords are not enough to deal with sophisticated attacks nowadays; and just like technology, the authentication process should also modernize itself for more innovative and secure financial transactions. Among all the techniques available, Voice signature authentication comes off as the easiest and most compelling solution taking the advantage of the user's individual uniqueness for quicker and secured user-friendly addition to the existing system.

In Bangladesh from the 1980s, people from all the country are continuously being introduced to the banking system, for rural communities in Bangladesh, where people hardly know how to write their names in such scenario the traditional system are already much complicated and so in this regard the idea of voice recognition play a great role to ease up their inclusion into the banking system. Now, on the other aspect, we see there is an ongoing mobile banking revolution in current Bangladesh and most of the security systems are based on password or PIN which are very prone to hacking and theft while other methods are more time consuming; the Voice signature authentication is much faster and reliable in this regard as well.

Voice signature authentication incorporates the uniqueness of individual vocal acoustic patterns which separates one from another. Each of these voices has unique features that help to identify it from others. These features are not limited to, pitch, tone and accent, with speech patterns that could be analyzed for corrupted or fraudulent features or AI-generated features using techniques under Deep Learning. Moreover, it is interesting to note that biometric systems are increasingly being pursued for integration in finance; among them, only voice signature authentication is non-intrusive and easy to conduct, thus being especially suitable for mobile and remote banking applications. Inherent properties of human voice make it possible to classify one from another. The human vocal system has individual unique structures that allows a person to be recognized and familiarized based on memory of previous interactions.

Furthermore, this study will address the challenges and future scopes with the adoption of voice signature authentication in financial systems. The analysis will cover

its aspects in terms of technical architecture, along with the challenges associated with voice data collection and preprocessing, implementation of effective models while overcoming the implications for user privacy and data security crucial to this sector. As financial institutions strive for flawlessness, convenience and user trust, it is essential to ensure that the proposed methods are implemented according to the needs of the financial sector.

The implementation of voice signature authentication techniques in financial systems propelled by the advanced Deep Learning opens up new doors not only for the financial transactions rather paves the way for the future of voice-based authentication in other sectors like customer service, smart homes, automotive systems, security access control, VR (Virtual reality) experiences, aeronautical-aviation industry and internet of things (IoT) [1].

1.1 Background

The evolution of digital technology has profoundly transformed the global financial landscape, enabling new forms of convenience, accessibility and automation through online and mobile banking systems. In developing countries like Bangladesh, the rapid adoption of digital financial services such as bKash, Nagad and Rocket, along with the digital expansion of traditional banking systems and financial institutions, has made cashless transactions and mobile-based money transfers a part of everyday life. Individuals across both urban and rural regions can now perform essential activities such as sending money, receiving salaries, and making payments directly from their mobile devices.

Despite these advancements, security and user authentication remain persistent challenges in digital banking systems. Many users, particularly in rural and semi-urban areas, face difficulties in remembering Personal Identification Numbers (PINs) or complex passwords required for financial transactions. Elderly individuals and people with limited literacy often rely on others for assistance, which exposes them to risks of fraud, identity theft and unauthorized account access. These vulnerabilities not only threaten financial security but also erode public trust in digital and institutional financial platforms.

In this context, there is a growing need for a more natural, reliable and user-friendly authentication system that ensures security without depending on written credentials or memory based codes. One effective solution is voice signature authentication, which identifies users based on the unique characteristics of their voice. Human speech inherently contains distinctive acoustic and linguistic patterns shaped by factors such as age, gender, regional dialect and articulation style. These characteristics are highly individual and difficult to replicate, making voice based authentication a practical and secure alternative to PIN or password based methods in financial systems.

Recent progress in deep learning has made it possible to analyze, classify and recognize voice patterns with remarkable precision. In this research, advanced architectures including a Modified BiLSTM, Hybrid Residual CNN, Deep CNN and a

Siamese network are integrated into a unified individual framework for voice signature authentication. Each component contributes distinct strengths: the Modified BiLSTM captures long-term temporal dependencies in speech sequences, the Hybrid Residual CNN enhances feature representation through residual learning, the Customized CNN extracts discriminative spatial features and the Siamese network measures the similarity between voice embeddings to verify identity. Together, these models form a comprehensive Deep Learning approach capable of learning high-level voice representations and performing accurate speaker identification.

This study, titled "Implementation of Voice Signature Authentication Techniques in Financial Systems Using Deep Learning," aims to design and develop a reliable voice-based authentication framework specifically tailored for financial environments. The proposed system leverages deep learning models to ensure accurate, consistent and secure user identification across various financial applications. By incorporating voice signature authentication into the operational workflow of banks and financial institutions, the framework addresses critical issues related to security, accessibility and trust in digital transactions.

Furthermore, the proposed system seeks to reduce incidents of fraud and misuse, enhance transaction integrity and improve the overall user experience. It offers a viable alternative for individuals who face difficulties using conventional authentication systems. Ultimately, this research envisions a more inclusive and secure financial ecosystem in Bangladesh, where a person's voice acts as their unique digital signature, ensuring safe, seamless and intelligent access to financial services.

1.2 Motivation

In today's digital era, where digital transactions in the financial system have become one of the regular activities, need for a secure, cost-effective, user-friendly and high performance authentication system. Traditional authentication systems like PIN, password and signature are mostly prone to data loss, data theft and data duplication, which are unable to ensure a secure authentication system in financial transactions. This limitation greatly motivates to development of a voice-based authentication system, which will ensure a secure yet user friendly authentication system in the financial system.

Voice based authentication technique stands out among all the existing authentication techniques because of the unique characteristics of the human voice, as it carries unique voice patterns, which are created by user's accent and speaking habits. These acoustic features make the voice-based authentication system more robust in distinguishing between an actual versus a fake speaker. Moreover, unlike the other traditional authentication systems, which require expensive hardware devices for executing the authentication process, voice authentication does not need costly hardware devices; rather, it can be implemented through a regular smartphone. This aspect of the voice authentication system is highly motivated by developers to build voice based authentication systems in digital transactions like mobile banking and mobile payment systems.

Furthermore, most existing voice authentication systems are based on the English language and the Bangla language remains relatively underexplored. Literacy barrier and less technological knowledge have restricted the Bengali speaking people from exploring the existing authentication systems, like text or password oriented. These addressed limitations require an accent aware Bengali voice authentication system, which will be able to overcome the limitations faced by the Bengali speaking people while experiencing digital transaction services.

Additionally, the recent advancement of deep learning models like BiLSTM, CNN and Siamese models has significantly increased the performance of the voice authentication system in recent years. These deep learning models are not only able to learn the acoustic features of the user's voice but also able to successfully identify the correct user. So, this motivates the research to incorporate deep learning models in the Bengali voice authentication system to sufficiently identify the authentic user voice.

Overall, the motivation lies behind the research in developing a system that aims to become a promising alternative to existing authentication systems, making a safe and user-friendly environment for financial interactions.

1.3 Problem Statement

In the modern perspective, most financial transactions must ensure authentic and smooth methods of authentication as the world is getting digitized and cashless day by day along with increasing cyber crimes. According to Bangladesh Financial Intelligence Unit (BFIU), in the fiscal year 2022-23; 14,106 reports filed for various suspicious transactions which is 65% more than previous year [2]. In such scenarios worldwide, voice signature authentication could solve these challenges by allowing real time verification by integrating NLP techniques in verification. In the following research problems along with the complications that will arise in the implementation of such authentication processes along with the potential solutions and proposed ideas to deal with such problems with more accuracy and less complexity are discussed.

Firstly, financial systems always handle a lot of sensitive personal and financial information. Since voice data is used for authentication, handling should be done with ultimate care without compromising performance. Though this task is quite challenging. It can easily be a prey of hacking or replay attacks using voice-synthesis technologies, to fake the voice of a valid user. Advanced encryption methods can involve homomorphic encryption on speech data that needs to form a part of processing the data with sensitive information not accessible to attackers. Again, deep learning based models with CNN and LSTM can be used as anti-spoofing to determine the synthesized portion in data.

Based on the papers [3] , [4] , [5] , [6] , [7] , [8] depending on physical and emotional cactors and also illness, age, stress or emotional states are some important factors tempering variability in features of an individual's voice that lower the accuracy of voice authentication models. These factors result in false rejections or false ac-

ceptances in financial transactions. Two possible ways to handle the challenge is by creating adaptive learning and transfer learning, which over some time, continuously learn and adapt to the voice patterns of a particular user. It would also be helpful if the system could include NLP driven modules of emotional and stress detection operating in collaboration. In summary, the result of all these would be better for recognition and compensation for the differences that come up.

Another important challenge in voice authentication is that big, good-quality and balanced datasets are not available labeling various accents, tones or gender, especially in financial systems. Most public datasets do not capture the range of variability of the real world conditions in the voices, making the generalization with models a bit more difficult. Opening datasets toward greater diversity through the generation of synthetic voice data using AI-based techniques such as Generative Adversarial Networks (GANs) can be a source of light to this problem [9]. These AI-generated datasets can mimic many different accents and emotional states and some have various noise environments all of which will contribute much to the models' performance. Specialized voice datasets that include wide demographic and linguistic variation in financial transactions will also be curated and will ensure system reliability in real-world scenarios.

The latest improvements to various AI-generated voice technologies, including deep fake voice synthesis, seriously threaten the security of voice identification systems by tampering with one's voice for fraud [8]. AI-generated voices can impersonate legitimate users and it will be tough to distinguish the difference between real and fake synthesized voices. Semantic analysis, Deep learning models with multiple variables like voice, face or fingerprint can be followed to reduce the fraud risk [6]. Adding to that is biometric multi-factor authentication, like voice with face or fingerprint biometrics.

Again, the financial institutions need voice signature authentication systems capable of handling volumes of huge transactions with users in real time. While it is true that deep learning models are fairly accurate, it is their very complexity that may introduce latencies large enough to make real time processing challenging [9]. Here high accuracy models with much lower complexity models like Lightweight CNN models and weighted quantization methods can be very useful [10]. Deployment on edge devices for real-time processing will even slice the latency because computations are offloaded from a central server to localized hardware.

Another thing to concern is that, users may use financial systems with devices differing in type, including mobile phones and laptops, and may be placed in different environments like quiet rooms and open public places. Different devices have different sets of microphone and noise cancellation technology which will create abnormality in the authentication and feature matching processes. Here, it is possible to adopt some cross device and cross environment training strategies in order to ensure robustness of the model. For better generalization, learning features with domain adaptation techniques can be implemented by overcoming variability and furthermore contextual based NLP models can evaluate the use case for high-risk transactions maintaining a threshold for authentication.

Other variants of attacks pose a threat to the integrity of voice signature authentication in financial systems; for example, voice mimicking, whereby a fraudster closely imitates the voice of a legitimate user. Integrating personal questions and random word pronunciation can help to mitigate these issues creating a randomness to the process for the hackers which then will be analyzed based on manners, style, accent, patterns, and choice rather than depending on only acoustics as these are far more difficult to mimic and adds an additional layer of security [9]. Furthermore, deep fake detection algorithms to differentiate between an imposter and desired speaker can be utilized on these patterns as these are for more difficult spoof adding multi layered voice authentication steps.

Another factor is to be considered as voice signature authentication needs to perform its authentication process across diverse regional accents and dialects and these dissimilarities in pronunciation or choice of words declines the overall accuracy of the authentication process. Using multi-regional datasets with transformer-based models Wav2Vec 2.0 can be adopted to make the model more comprehensive; for which Deep Learning techniques in tuning the input without losing features can give better results [4], [5], [11].

Financial voice authentication systems operate, in most cases, under conditions of variable background noise. Models existing will likely not perform well or totally fail for environments characterized by heavy background noise, such as a busy office or outdoors. Deep learning models for data augmentation and spectral analysis can be taken into consideration for robustness and auto encoding the input signal overcoming the noise levels [5], [7], [12]. The integration of such with NLP models that grasp structural aspects of speech in noisy environments certainly guarantees more reliable authentication.

User unfriendliness is an unavoidable concern as forcibly asking the users for repeated authentication owing to false rejections frustrates the users, especially in high-at-stake environments such as banking. The trade-off between security and user convenience must be met. The enhancement of an adaptive learning capability of voice models allows voice models to learn the individual variation in voices over time, which consequently allows it to lower the false rejection rate. The same goes for the utilization of Deep Learning in judging semantic consistency between spoken content and expected text, again ensuring system reliability without excessive re-authentication. [4]

So the basic idea that this research is trying to explain is :

How effective is the combination of multiple independent deep learning models such as age-gender classification, AI-generated voice detection, dialect identification and Siamese-based speaker authentication in Bengali voice signature authentication in financial systems?

Integrating multiple deep learning models that operate independently on distinct aspects of human speech can substantially enhance the reliability and precision of

voice signature recognition systems. Each model contributes specialized insights into different speaker characteristics, allowing for a multidimensional understanding of vocal identity. The age-gender classification model captures demographic acoustic features such as pitch, timbre and vocal tract length, which provide essential cues for distinguishing between speakers with similar tonal qualities [13]. Additionally, incorporating an AI-generated voice detection model strengthens the system's resilience against spoofing and synthetic speech attacks are a growing concern with the rise of generative voice cloning technologies [14]. Meanwhile, the dialect identification model analyzes regional phonetic variations and linguistic patterns, which have been shown to improve recognition accuracy by accounting for sociolinguistic diversity [15].

This model leverages spectro-temporal features and convolutional architectures to identify subtle inconsistencies in synthesized speech that are often imperceptible to humans. Finally, a Siamese-based speaker authentication model provides the core identity verification mechanism by comparing voice embeddings and learning discriminative features between genuine and impostor samples [16]. By combining the outputs of these independently trained models, the system achieves a layered defense mechanism and a richer representation of speaker characteristics. This ensemble like framework ensures that contextual, demographic and biometric traits are all considered, leading to improved performance under varying acoustic conditions and across multiple recording environments. Moreover, the modular design enables flexible deployment and efficient scalability; each model can be updated or fine-tuned independently without retraining the entire system. Thus, the collaborative use of multiple specialized deep learning models not only enhances the accuracy and adaptability of voice signature recognition but also increases its resilience against adversarial and environmental challenges [17].

How do traditional feature extraction techniques such as GMM (Gaussian Mixture Models) compare with newer deep learning-based models (e.g., CNNs, RNNs) in terms of accuracy, robustness to noise, and adaptability in voice signature authentication for financial systems?

Even though CNN, RNN and GMM have some advantages as well as disadvantages compared to other models when dealing with voice speech recognition, at the end of the day it works quite similarly in those three models. GMM (Gaussian Mixture Models), which have low computational complexity and give quite good results on small datasets. Since GMM is based on Gaussian distribution assumptions in modeling the complex voice patterns and noisy environments there remains a significant challenge for this model for accurate results. In turn, CNNs have a greater native capacity to feature extract than spectrograms (when highly accurate), as discussed above. CNNs also seem to be very noisy creatures (even more so when trained on noise-augmented datasets). Well, RNNs are proven to be very good at modeling temporal dependencies if we see that they would closely resemble sentences when dealing with something like sound wave or any kind of continuous voice signals. Moreover, RNNs are even more computationally expensive than CNNs in general because of the need for sequential input causing issues to on-time usability. The paper [18] suggests that, GMMs are a fast and analytically tractable algorithm that

sacrifices some accuracy for efficiency in processing, CNNs trade-off between the RRN level of noise robustness on one side and higher levels of both trace-acyclicity or complexity (best when less relevant) error rate will be highly resource requirements; all had also shown better results than would have been possible as well). In this case, the choice will depend on whether a particular financial system application has more interest in accuracy, computational efficiency or handling of temporal data.

This research will examine the overall implementation of these comparisons and find a suitable method for the voice signature authentication in the banking system.

1.4 Objective

The primary objective of the research is to build a Bangla Voice Authentication System using Deep learning techniques in the financial system to accurately verify the identity of the actual speakers. This aims to develop a consistent and efficient voice-based authentication technique, which is considered a reliable alternative to traditional text based, PIN or password based authentication techniques. Since most of the existing voice based authentication systems are mostly English or other globally common language-based, introducing the Bengali voice-based authentication system points out the lack of linguistic diversity in the traditional financial systems.

To accomplish this goal, the first objective is to collect a Bengali voice dataset that includes voice data from different age groups, genders and linguistically diverse accents. The next objective is to preprocess the collected dataset for the next step. The preprocessing stage includes cleaning, splitting, noise handling, standardization, feature extraction and normalization of the collected data. Each voice data is standardized in 5-second audio clips. Crop striding method, which crops the longer recordings and pads the short recordings, helps to add diversity in the dataset and makes the system scalable in real-world applications. Noise augmentation helps to add noise in the data at varying SNRs to generate noise samples of different categories, like urban, indoor and outdoor. The next step is relevant acoustic feature extraction, which possesses unique characteristics of the user's voice. In the following step or in the model training step, these features are used as input to train different deep learning models. Several deep learning models. This research focuses on deep learning model architecture, like a Siamese model helps to find similarities between users. Age, Gender models help to find the age, gender of the speakers and an AI model helps to find AI-generated fraud voice. Moreover, a CNN model is trained for identifying regional dialects. This process of training the models helps identify the correct user with high precision and efficiency. Next, in the testing phase, the result of the models is evaluated through accuracy, recall and precision metrics, which show the effectiveness of the developed system. Along with the technical accuracy, the research prioritizes reliability, usability and Scalability to ensure the system works efficiently in diverse environments and also incorporates the system with digital transaction services like digital banking or mobile app transaction services.

Overall, the primary impact of the research is expected to develop a good-quality

dataset, train the models to differentiate actual vs fake speakers, and measure the accuracy of the performance of the developed system. Ultimately, the research aims to develop an efficient and high-performance Bengali voice authentication system, which ensures security and accuracy in real-world digital transaction applications.

1.5 Methodology in Brief

The research follows a structural methodology that includes several systematic processes starting from data collection, following preprocessing, model training, and ending with testing to develop a reliable and effective Bangla voice Signature Authentication system. The first stage involves Bengali voice-based dataset collections from multiple sources. This attempt to collect data from several sources ensures diversity in the dataset, covering variation in age, gender and accent to make a reliable and robust voice verification system. The next step is preprocessing the data, which includes data cleaning, standardization and balancing the raw audio data for model training. Each voice data is standardized to 5-second audio clips. Crop striding method is being used to crop the longer audios into 5-second audios, and smaller ones are padded to get into the standard size. Once the standardization is done, noise augmentation is incorporated into the system for identifying the speaker's voice in noisy backgrounds. Then, through the acoustic feature extraction process, unique characteristics of the users are extracted, which represent both static and dynamic features of the speech. The combined pre-processing steps ensure clean, consistent, and high-quality inputs for the BiLSTM-based classification model and the Siamese MLP verification model, which ensures the high accuracy performance of the system.

Next, Deep Learning models like Siamese, Age and gender classification model, AI model and CNN BiLSTM model are being trained to provide specific outputs like detecting similarity between users, age and gender detection, artificially generated voice detection, and predicting dialect class accordingly. Each model combines different layers of neural networks to learn and extract key features of the speech data. The gender classification model uses a BiLSTM network along with a shared encoder. The shared encoder processes the input data to learn the general patterns that are common to all the speakers. Then, the BiLSTM layers used the encoded data to analyze the sequence of the speech both in the forward and backward directions. A generalized Gender attention layer plays one of the most important roles, identifying the gender difference. After that, several dense layers are used to predict the gender of the speaker, whether male or female.

The age classification model follows the same approach as the Gender Identification Model, where the BiLSTM and shared encoder are used. The shared encoder helps to learn similar patterns and the encoded data is passed to the BiLSTM layers to analyze the sequence in both directions, forward and backward. An age attention layer has been added to find the patterns that change with age, such as pitch and tone of the voice. This mechanism helps the model to be more age sensitive. After the attention layer, the features are passed through dense layers, which identify different age groups.

The Regional Dialect model is developed to classify speakers' regional accent by com-

bining a BiLSTM and a CNN model. By incorporating hybrid models, the model learn the local sound pattern from CNN and long-term dependencies from the BILSTM model. Initially, CNN layers extract acoustic features like energy, pitch, and frequency variation and after extraction, these features are passed to the BiLSTM layers to find the sequential relationships in the speech of the users. A dropout layer is added to prevent overfitting and finally, a fully connected dense layer identifies the speaker's regional Dialect class.

The AI model uses a CNN architecture that helps to learn acoustic feature extraction from the input data, which is fed into it in the form of MFCC or spectrogram. Then, these data are passed to the Convolution layer, RELU, and max pooling layer. These layers help to find sound patterns and reduce noise. Dropout layer works for avoiding overfitting, and Feature flattening is done in the flatten layer. Lastly, the Sigmoid layer identifies the real and fake speakers, also determines whether the speakers are already registered or not.

The Siamese model is designed to compare two speakers and identify whether the two speakers are the same or not. For this, first, two input features are processed through two identical networks in parallel. Each network then produces an embedding vector through a shared dense layer. These produced embedding vectors are transferred to the similarity module for similarity check and then passed to the similarity classifier or MLP, which creates a combined feature vector that is finally transferred to the sigmoid classifier. The sigmoid classifier produces a similarity score for correctly identifying whether the two speakers are similar or not. Then, the models are being tested to show the performance and accuracy of the model by the precision, recall, loss, and validation matrices. This test phase determines whether the model is efficient enough for implementation in real-life digital transactions or not. As illustrated in Figure 1.1, the proposed framework employs multiple independent deep learning models that analyze different aspects of the input voice signal—such as age, gender, dialect, and authenticity—to enhance the robustness of speaker authentication.

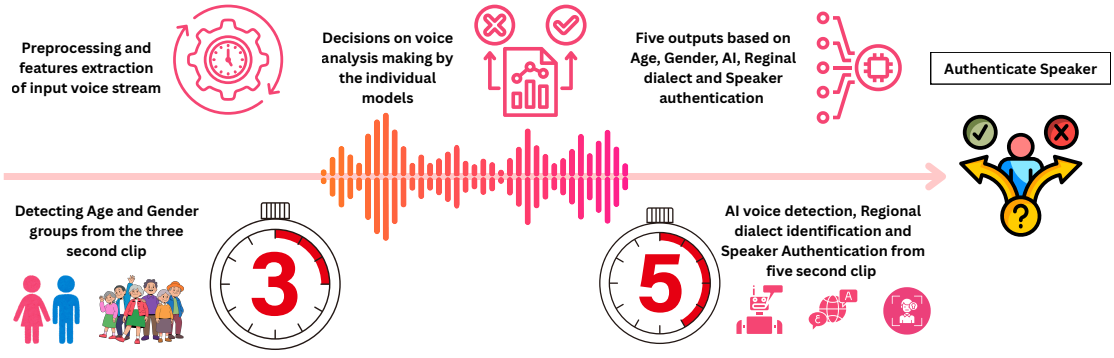


Figure 1.1: Overview of the proposed pipeline in speaker authentication.

Overall, the implemented methodology establishes blended data-driven techniques, language-specific adaptation, and deep learning models to build a robust, secure, and user-oriented system in digital financial applications.

1.6 Scopes and Challenges

The scope of the research encompasses the design, development, and implementation of the Bangla Voice Signature Authentication system, which focuses on verifying the different accented Bangla speakers to authenticate their identity in the digital transaction system, like digital banking or mobile app money transfer. This research emphasizes the utilization of Deep Learning models for developing Bangla-speaking Authentication, which has been less explored in the voice-based authentication domain. Moreover, the correct implementation of the system not only makes it economically beneficial but also encourages them to get involved more in digital transactions.

However, along with the various scopes of the research, several challenges can be faced during the development of the system. The first and foremost challenge is collecting the Bangla voice dataset, which is publicly less available. That demands a systematic way of collecting data and pre-processing the data to add sufficient variety to the dataset, representing different ages, genders, and regional accents.

Another challenge is, the models have been trained with a relatively small dataset, which may result in not being able to process and handle large-scale user requests. Lightweight models have been designed to handle the limited Bengali voice data, which may impact the model's ability to work effectively on unseen voice data. So, the models may not be able to run smoothly under extremely heavy user requests.

One more major challenge observed in the siamese model is the tradeoff between precision and recall. When the recall rate increased to above 90%, the precision rate dropped to around 80%. Inversely, when the precision rate approaches around 100%, the recall rate falls around 3%. Even when the precision rate is adjusted to 95%, the recall rate remains at a very low level, approximately 25%. This pattern indicated that the imbalance between high recall or correctly identifying the actual speaker and high precision or minimizing false identification remains a challenging aspect of the developed system.

Chapter 2

Literature Review

Today, there is a growing focus on changes in financial systems because of the need for better protection in an increasingly digital world. As consumer interactions and transactions move through electronic channels, the need for clear and reliable methods to verify these transactions has become important. One technology in this area is voice signature authentication, which uses deep learning to provide secure and accurate voice-based verification [19]. As more smart devices can capture and process voice data, these systems not only make interactions easier but also address some of the problems of traditional methods [20]. By analyzing unique voice features such as pitch, tone and speaking style, deep learning based voice authentication offers a simple and effective way to verify users. A key question arises, how can deep learning be used in daily life to make voice authentication smooth and reliable? This approach allows systems to adjust to different accents and speech styles, making financial and other services easier to use [21].

2.1 Deep Learning

Deep learning, situated at the intersection of computer science and machine learning, is revolutionizing human-technology interactions by enabling machines to comprehend, interpret, and generate information from raw data. This paradigm employs artificial neural networks to process data and solve complex problems in a manner akin to human understanding [22]. Applications of deep learning span various domains. In natural language processing, models convert spoken language into text, recognize images, summarize lengthy documents and build question-answering systems. For instance, speech recognition models utilize deep learning to transcribe audio input into text, while other models detect patterns in data or analyze user feedback for sentiment [23]. In the financial sector, deep learning has been instrumental in monitoring market trends, enhancing automated customer service systems, detecting anomalous patterns to prevent fraud, and supporting voice-based authentication systems [24]. Over time, these models have improved in understanding complex data, adapting to variations in inputs, and delivering accurate results across diverse tasks and applications [25].

2.2 Automatic Speaker Recognition (ASR)

Automatic speaker recognition (ASR) is the technology used to extract features such as frequency, pitch, speaking rate, and energy distribution from speech and to classify each speaker based on their unique voice characteristics [26]. Deep learning models—including recurrent neural networks (RNNs), convolutional neural networks (CNNs), and more recently transformer-based architectures like Wav2Vec2.0 and Transformer allow ASR systems to learn features directly from raw audio, reducing reliance on manual feature engineering [26, 27]. RNNs and LSTMs are especially effective for sequential data, while CNNs and spectrogram-based methods help with phoneme and word recognition across different accents and variations [28]. In recent work, Conformer-based models adapted via transfer learning and knowledge distillation have achieved very low equal-error rates in speaker verification tasks [29]. Other studies show that speaker-attributed training using multi-stage encoders and attention-weighted speaker embeddings improves ASR performance in overlapping and multi-speaker environments [28]. Transformer models with innovative pooling approaches such as Temporal Gate Pooling also enhance speaker identification accuracy with fewer parameters [30]. In financial systems, ASR plays a key role in voice signature authentication; with these advances, modern systems can more accurately understand spoken commands, recognize speakers reliably, and provide secure, user-friendly authentication [29].

2.3 Related Works

The field of speaker recognition and speech processing has witnessed significant evolution over the years, moving from classical statistical approaches to modern self-supervised and multimodal frameworks. This section reviews relevant works in a chronological and thematic order, highlighting the gradual progression of ideas, architectures and applications.

2.3.1 Classical and Foundational Approaches in Speaker Recognition

Early research in speaker recognition primarily focuses on statistical and signal processing-based models. The work in [31] implements a Bidirectional Long Short-Term Memory Recurrent Neural Network (BiLSTM-RNN). Feature extraction is done using Mel Frequency Cepstral Coefficients (MFCC) on the filtered audio and then dimensional reduction is done with Probabilistic Principal Component Analysis (PPCA) to decrease run time. Yasunari Obuchi's (Carnegie Mellon University) pitch detection algorithm (PDA)'s speech dataset is used. The model is trained with 250 samples for each 9 speakers. The proposed model obtained 95.77% accuracy and 94.29% sensitivity, which is better than PRNN and KNN models. Similarly, the study in [32] compares the performance of the model Gaussian Mixture Model-Universal Background Model (GMM-UBM) with CNN for text-independent closed set automatic speaker identification and DL-based techniques like Factorized Time-Delay Neural Network (FTDNN) on two a datasets- "Call Center" a dataset prepared by the researchers themselves totaling four hours of speech after removing silent segments, which is considered limited data for ASI tasks. The other is a subset

of "LibriSpeech" that includes 100 hours of speech data from 251 speakers. Moreover, with abundant data (LibriSpeech) the proposed CNN Model achieves 99.5% accuracy on only 1-second utterances. The researchers did also conclude that for any new enrollments, GMM-UBM is undoubtedly easier to manage and lower cost, thus more practical for cases with a limited number of users but requiring faster decision making.

The research in [9] extends traditional feature engineering distinguishing between micro and macro level features where micro being more volatile and easily manipulated whereas macro level features contain more information but are difficult to extract. This paper proposes an idea of extracting aggregated macro-level features using the FRESH (feature extraction based on scalable hypothesis test) algorithm from the MFCC augmented series. LPCC, PLPC, MFCC, DELTA techniques are used to determine which feature is to be extracted for better outcome of the results. The dataset from IIT-M IndicTTS and VoxForge is used here with an accuracy of 99.94% and 98.43% respectively. Cross-lingual studies, such as [33], experiments on the speaker profiling dataset The National Institute of Technology Karnataka's to check the performance of several models for cross lingual speaker identification. The models are tested in their native languages once by training with English recordings of the same person, then training in the same language as the test. It is found that LSTM and ResNet-50 performed best overall for both cases except Malayalam by training in English.

2.3.2 Advancements through Deep Learning Architectures

With the advent of deep learning, researchers begin focusing on more expressive architectures capable of capturing complex temporal and spectral dependencies. The work in [34] identifies redundancies in learning parameters in the hidden layers of Mockingjay, a BERT-based model. They introduce exploring tasks to analyze knowledge encoded in pre-trained networks that assess the information richness of each layer's representation, evaluated through their performance in downstream tasks. To remedy this, they proposes Audio ALBERT, a self-supervised model designed for speech tasks implementing the concept of Parameter sharing from ALBERT(A Lite Bert). For downstream tasks such as speaker recognition, this yields similar results to other pre-trained models(transformers), but with much smaller networks. In the internal layers of Audio ALBERT, the representations encoded contain more information related to both phonemes and speakers compared to the last layer, which is typically used for downstream tasks while having 91% fewer parameters compared to other massive pre-trained networks. Similarly, for identifying multiple speakers on the same channel (co-channel speaker identification) this paper [35] proposes an end-to-end system based on methods such as residual attention, dilated convolution with residual connections, Siamese networks for speaker identification, and a novel max-pooling error metric. The test data is extracted from "Hub-4" broadcast news recordings. ResNet and Syllable-level BI-LSTM models are used and the prior is found to be the overall best detecting 3 out of 3 speakers from a channel 81.2% of the time.

Further, studies like [5] discuss an idea of transfer learning based multi-channel trans-

former speaker recognition system overcoming the support vector machine (SVM), Gaussian mixture model (GMM) and i-vector models and their complications. Initially, from the main signal, two input channels of harmonic and percussive are separated by HPSS and then by CNN method and LeViT architecture these two along with the original are fused to get better improvement in the signal. Here, the key part is that to overcome the complexity of the dataset size, this paper adopted Transfer based schemes of converting audio signals into spectrograms which is then used to verify speakers overcoming the need for a larger dataset. Based on the dataset VoxCeleb1, the proposed idea achieves an accuracy of 94.94% (STFT only) and 98.94% (DSS II). The review in [7] basically addresses the main challenges like accent variability and noise faced in speech processing, for instance in - ASR, speaker recognition and synthesis. According to the authors the deep learning models such as- RNNs(LSTM,GRU), CNNs(2D,1D), transformers, and conformers are now capable of automatic feature extraction, as a result, the performance improved a lot. On the other hand, the traditional models like- HMM-GMM lacked robustness and required manual feature engineering, which is solved by deep learning models now.According to this paper, the state-of-the art Conformer model achieved 1.4% in case of WER metrics on a clean dataset named 'LibriSpeech'. Speaker recognition accuracy is evaluated using EER, with VoxCeleb widely used.The naturalness and clarity of synthesized speech is improved by models like FastSpeech and Tacotron and we can know that by evaluating the Mean Opinion Scores. In addition to this, using metrics like f1-score and AUC, it is observed that the Conformer-based VAD models performs better than other models, excelling in noisy conditions. A newly proposed probabilistically weighted SVM model is suggested by the authors of [10] in identifying speakers from multi-speaker speeches and surrounding noise in real time by overtaking the hurdles of speech treatment. Firstly, SVM and HMM (Hidden Markov Model) emotional class is predicted and then for structural features double sparse learning (DSL) is used, along with these aggressive and spectral models are used to produce relative weighted acoustic and linguistic features based on different frequency and speech signals. Here stochastic approach and template-based approach are defined to convert the signal to standardized type for better rate of recognition. This proposed method ensures up to 86% rate of recognition for the dataset CSTR-VCT (Center for speech technology research - Voice cloning Toolkit) over the existing methods like PCA (64%) and BPNN (70%). More recent work such as [36] mainly focuses on the challenges of verifying speakers based on short, non-English audio clips, using deep learning. The researchers here used recordings of Ukrainian politicians and by using this dataset, they compared different models for example- TitaNet, WavLM, PyAnnote and ECAPA. And to get the results, they mainly focused on metrics like- Equal Error Rate (EER), False Rejection Rate (FRR) and False Acceptance Rate (FAR). According to ECAPA and TitaNet stood out with the highest accuracy, with EERs at 1.71% and 1.9% . After analyzing it is found in that, PyAnnote model performed well in case of both accuracy and quick processing. On the other hand, the WavLM model couldn't perform well compared to other models.

2.3.3 Self-Supervised and Transfer Learning Approaches

As data availability and computational power expand, self-supervised and transfer learning methods gain prominence. It is seen that large scale unlabeled data gives out better generalization than supervised learning which led to a great interest in this field in recent times. The research in [6] tries to explore the different aspects of self-supervised learning compared to the existing techniques. Wav2vec 2.0(XLSR), UniSpeechSATLarge and HUBERT are decided by the authors, as these learning methods don't rely much on assumptions and cause minimal architectural and fine-tuning changes. Here, a weighted average of representation of all layers is used rather than just the final layer of pre pre-trained models to implement the models. About 30% relative improvement is obtained over others (Fbank, MFCc) by the proposed idea, even outperforming the winner for the VoxCeleb 2021 evaluation set. Similarly, in [37], there is a discussion about the way of improving spoken language models by the use of transcripts from automated speech recognition (ASR). The FlauBERT model is moderated by the author to generate FlauBERT-Oral which is a language model, by undertaking the conversion of 350,000 hours of French radio and TV content. Their research demonstrates that even with its noise, ASR-generated text enhances spoken language modeling. The model demonstrated the possible use of ASR-generated data for language modeling by surpassing previous models in syntactic parsing, TV program classification, and word prediction.

Building on these ideas, the research paper [38] explains a challenge in spoken language understanding (SLU) and spoken question answering (SQA), where current self-supervised speech encoders, like Wav2Vec 2.0 and HUBERT, mainly capture auditory features rather than the deeper semantic meaning necessary for tasks like intent named entity recognition (NER), classification (IC), slot filling (SF), and SQA. To solve this issue, semantic knowledge is introduced into speech encoders by BART, which is one of the large language models (LLMs), by using the "Semantics into Speech Encoders" (SSE) method in [38]. And as a result of this, the task's performance improved a lot without the requirement of labeled audio transcriptions. And after analysis it is found that, SSE performs better than other mentioned models, including-63.64% accuracy in SLURP-IC, 80.10% F1 in SLUE-NER, and 57.2% in NMSQA datasets. Moving on, the authors of the paper [39] discusses the challenges such as the struggle with handling varied tasks and not using dialogue context by the current speech-text pre-training models. To solve these issues, a speech text dialog pre-training model named SPECTRA is designed to better capture cross-modal alignment and multi-turn context in conversations. According to the authors some innovative pre-training tasks such as- Temporal Position Prediction and Cross-modal Response Selection, are engulfed in the SPECTRA model. The model is trained on the Spotify 100K dataset, which gives it access to speech-text alignment and dialog history for improved comprehension.According to this paper, SPECTRA model performs better and showed more improvements, comparing with other existing models and achieved high accuracy in a number of tasks such as- 87.50% in Multimodal Sentiment Analysis, 67.94% in Emotion Recognition, 73.48% in Spoken Language Understanding, and 21.96% in Dialog State Tracking.

2.3.4 Applications of Speech Models in Real-World Systems

The evolution of speech models also drives advancements in diverse real-world applications. For instance in [40], two techniques called 'pipeline' and 'end-to-end (E2E)' for Named Entity Recognition (NER) from voice are being compared. A two-step process is often being followed to achieve NER from voice. Performance has, however, greatly increases due to recent developments in neural models, particularly in bidirectional Long Short-Term Memory (biLSTM) networks. The work improves accuracy over prior systems, such as the ETAPE campaign results, by introducing a 3-pass pipeline architecture to handle tree-structured named entities. Furthermore, the study examines E2E models, demonstrating an increasing interest in this strategy even if the pipeline technique continues to be better for structured NER tasks. Fairness and inclusivity concerns are addressed in [4] and [41], where the research work [4] proposes addressing predictive bias in Automatic Speech Recognition (ASR) systems, which demonstrate performance discrepancies across demographic groups, especially those speakers who are marginalized, with various dialects, accents and genders. The research emphasizes how traditional evaluations using Word Error Rate (WER) ignore disparities in accuracy, which might have adversarial effects on community in intense situations. With the aim of understanding user interactions, the authors offered utilizing methodologies like intersectional benchmarks, qualitative error analysis and Human-Computer Interaction (HCI). By incorporating technical, dialectology and HCI perspectives, the research study attempts to boost ASR fairness and usability for marginalized populations. Similarly [41] discusses about the presumptions that commercial Automatic Speech Recognition (ASR) systems have biases against underrepresented groups, including speakers of accents or dialects other than the conventional kinds, such as Scottish or African American English. These algorithms tend not to correctly identify speech from underrepresented speaker groups due to a deficiency of diverse training data. In context sensitive techniques, both qualitative error analysis and numerical indicators are being highlighted for evaluating ASR systems. Using a project called "Lothian Diaries" which has diverse audio recording, this case study emphasized remarkable performance discrepancy over various linguistic groups. The authors suggest using user friendly evaluation methods to overcome presumptions and eventually improving the efficiency of ASR for different groups. The review in [3] further addresses the challenges in voice biometric systems, particularly in banking and security, by reviewing 50 high-quality research articles where the main concerns are the vulnerabilities in live detection health effects on voice, and challenges in differentiating between male and female voices. Methodologies for instance the tree-dependent clustering. For the purpose of improving system potency and dependability in diverse contexts, proposed solutions here concentrate on enhancing feature extraction, leveraging strong algorithms, and improving real-world applications.

In practical implementation like [11] includes an automated voice chat program named Shohojogi that is designed to refine banking in Bangladesh. It focuses on the issue of limited accessible, effective, and user-friendly banking service in Bengali language, that hinders financial inclusion and customer interaction. Researchers used the Bengali Common Voice dataset for speech-to-text conversion and for text summarization they used BANS dataset. They employed different models like the

gTTS library for speech recognition and a seq2seq model for generating text summaries, Doc2Vec model with BNLP toolkit's similarity measurement was utilized for effective tokenization and embedding Bengali text whereas gTTS library and seq2seq model was employed for speech recognition and generating text summaries respectively. The transcription of spoken Bengali into text was found to be fairly accurate with a Even though Character Error Rate (CER) was 0.06543, after using several question formats, overall accuracy of the system reached 70.001%. More of the system's vocabulary with diversified dialects incorporated by user feedback can improve the outcome as suggested by the paper. Similarly [12] examines employing voice biometric systems to enhance verification and security in banking by addressing the obstacles incorporating the need for more secure methods, surrounding noise, voice variations, language barriers and user admissibility. The research inspects a set of data that are taken from 50 distinct articles, evaluating speech demonstrations and their efficacy in various contexts. Machine learning models and biometric authentication models are utilized to verify and identify users in mobile applications as well as in the call centers of banking. Moreover, to enhance system performance and lessen the impact of noise, technologies like modern artificial intelligence (AI) and versatile learning algorithms are recommended in this paper. The study emphasizes high precision in real-world applications to improve customer satisfaction and security. It proposes solutions comprising strong testing protocols as well as constant adaptation. This positions voice biometrics as a crucial tool for following digital banking verification and fraud prevention. Meanwhile, the research in [42] aims to generate multiple models out of which they selected the best one and performed linguistic analysis on them to assess its gender and accent correctness while assuring the model's neutrality. A perfect speaker identification system should have minimal intra and inter speaker variance, respond correctly to imitation, and readily quantifiable attributes. There was no gender bias as the model was only marginally better at identifying female voices. However while testing for accents, the model showed discrepancy with one accent being the most predicted and another one being the least predictable thus highlighting the accent to be a challenge for Speaker Identification.

2.3.5 Limitations and Ethical Dimensions in Speech Verification Systems

Despite impressive progress, challenges remain concerning fairness, interpretability and generalization. The study in [8] suggests about confronting the challenge of speaker verification, especially in distinguishing if two utterances engaging those of role-playing agents belong to the same speaker. To generate balanced paired utterance sets categorized as "positive" or "negative,", the authors utilize diverse datasets drawn from popular media, such as Harry Potter, Friends, The Big Bang Theory, as well as the particular characteristics and conversations from therapy sessions. Methodologies include merging style-based models such as LIWC and authorship attribution models like RoBERTa. There are other methodologies like Sentence-BERT, fine-tuned with a contrastive loss objective function, annotators with various LLMs, Evaluation metrics encompass etc. In this paper, the accuracy for base, hard and harder levels were found 98.05%, 95.02% and 78.02% respec-

tively using the different models. The results indicate that accuracy in speaker verification tasks can be developed by blending stylistic and content-based features, which points that potential improvements in LLMs are enhanced for identification of speaker subtleties in individual statements. The findings in [42] further emphasize that accent and socio-linguistic diversity remain open issues. Together, these works highlight the growing awareness of ethical and social considerations in speech systems, suggesting that future research should balance technological advancement with fairness, inclusivity and transparency.

A concise overview of the reviewed studies is presented in Table 2.1.

Ref	Method/Model	Dataset	Key Results
[31]	BiLSTM-RNN, MFCC+PPCA	PDA speech dataset, 9 speakers	Accuracy 95.77%, Sensitivity 94.29%
[34]	Audio ALBERT	Mockingjay layers	91% fewer parameters
[35]	ResNet + Syllable BI-LSTM, Siamese Network	Hub-4 broadcast news	Detect 3 speakers per channel
[40]	End-to-End bLSTM	Voice data	3-pass architecture
[4]	Bias analysis in ASR	Multilingual speakers	Highlights WER bias
[32]	CNN, GMM-UBM, FTDNN	Call Center	CNN: 99.5% accuracy
[41]	ASR evaluation on under-represented groups	Lothian Diaries	Highlights performance gaps
[9]	Macro-level feature extraction (FRESH)	IndicTTS, VoxForge	Accuracy 98.43%
[5]	Transfer-learning multi-channel Transformer	VoxCeleb1	Accuracy 94.94% (STFT), 98.94% (DSS II)
[6]	Wav2Vec2, UniSpeechSAT	VoxCeleb1	30% relative improvement over baseline
[37]	FlauBERT-Oral, ASR-based LM	French TV/Radio transcripts	Improved spoken language modeling
[11]	Seq2Seq	Bengali Common Voice	Accuracy 70%
[38]	SSE + BART	SLURP, SLUE-NER, NMSQA	Improved QA accuracy
[7]	RNN, CNN	LibriSpeech	WER 1.4%, Improved speaker recognition
[39]	SPECTRA (Speech-Text Dialog Pretraining)	Spotify100K	Sentiment 87.5%, Emotion 67.94%
[10]	Probabilistic SVM + HMM + DSL	CSTR-VCT	Recognition up to 86%, outperforming PCA and BPNN
[33]	LSTM, ResNet-50	NIT Karnataka dataset	Best cross-lingual performance except Malayalam
[8]	Sentence-BERT	Media and therapy datasets	Accuracy 95.02%
[36]	ECAPA, TitaNet, WavLM, PyAnnote	Ukrainian politicians	EER 1.71%-1.9%, ECAPA andTitaNet best

Table 2.1: Summary of recent research on speaker recognition and ASR techniques.

2.4 Summary of Key Findings

The review of existing literature on the voice-based authentication system indicates that the system has become a reliable and secure alternative to the existing traditional systems like PIN, Password and signature-based authentication, when it is incorporated with Deep Learning techniques. Since most of the traditional authentication systems are prone to loss, theft, or duplication, in contrast, the voice-based authentication system assures that the data is not lost or duplicated, as it depends on the unique characteristics of the user's voice. With the rapid advancement of deep learning, the voice authentication system indicated it could be a promising alternative to existing authentication techniques for long-term use in the financial system.

The study shows that models like CNN, LSTM and BiLSTM can effectively analyze the unique characteristics of voice, such as pitch, tone, formant frequency and energy, which vary across speakers and remain consistent over time. The rightful utilization of these models helps make the system robust and the correct identification of the speakers under diverse environments, like a noisy environment.

Recent research has highlighted the advantages of Deep Learning, which enables models to understand speech and perform feature extraction. Models like BERT, Wav2Vec 2.0, and transformer-based architectures have enabled the identification of both the speaker and the meaning of the speech. These findings indicate that the combination of audio feature extraction and language modeling has made the voice authentication system more reliable and context-aware, and efficient in verifying the actual speaker, even if the voice is distorted and imperfect.

Additionally, data processing techniques that include data cleaning, balancing, standardizing, noise reduction and data augmentation have been proven to be one of the most efficient approaches for enhancing accuracy of the system's performance and Feature extraction techniques which include MFCC, jitter and shimmer have shown strong efficiency and high discriminative power in the voice authentication system identifying unique characteristics of user voice like pitch, tone and formant frequency.

Overall, the research indicates that combining deep learning based feature extraction with acoustic feature modeling can help develop a secure and reliable voice authentication system. This motivates the incorporation of deep learning techniques with acoustic analysis in the proposed system within the Bangladeshi financial system, like banking or mobile payment, which enables users easy accessibility and a secure user experience.

Despite having many advancements in the field of voice-based authentication systems with deep learning, which significantly improved the performance of voice recognition accuracy, there remain some key research gaps that are still unaddressed. Some of these are:

- **Unavailability of the Bengali Dataset:** Most of the voice authentication systems are based on English or other global languages. There is a very low

quantity of well-annotated and diverse Bengali language datasets available for voice authentication systems.

- **Limited Exploration of Deep Learning in Bengali Voice Authentication:** Advanced deep learning models like CNN, LSTM, and BiLSTM have been evaluated in many voice recognition systems, mostly based on the English language. However, in the context of the Bengali language, exploration remains limited. Therefore, there is a need for strong implementations of deep learning models such as Siamese networks or hybrid CNN–BiLSTM architectures for developing Bangla-specific voice authentication systems.

- **Lack of Integration Between Acoustic and Linguistic Features:** The Bengali language is characterized by phonetically rich and emotionally vibrant words, with variations depending on regions, dialects, and social settings. Since most existing voice authentication systems focus only on MFCC features, other unique characteristics that play a significant role in distinguishing speakers are ignored. Therefore, there is a clear research opportunity to develop an emotion-driven voice authentication system that performs accurately and efficiently in diverse speech conditions.

- **Limited Research on Lightweight Architectures:** Most voice authentication systems rely on heavy deep learning models, which require high computational resources and powerful GPUs, making them unsuitable for low-resource environments. Hence, there is a need to focus on lightweight models such as Siamese networks or compressed CNN–BiLSTM architectures that can run on everyday devices while maintaining high accuracy in voice-based authentication services.

In summary, addressing these research gaps will pave the way for developing an efficient, inclusive, and resource-friendly Bengali voice authentication system, advancing the field toward broader linguistic and real-world applicability.

Chapter 3

Requirements, Impact and Constraints

3.1 Final Specifications and Requirements

The final specification and requirements of the recommended system have been thoughtfully designed, considering both technical and non-technical limitations, following a comprehensive analysis of the problem area, evaluation of existing techniques, and significant feedback from stakeholders in the relevant field. The system is designed in such a way that both the functional and non-functional requirements are equally addressed. Functionally, the system includes processing raw input voice data through a clearly defined pipeline, which ensures data preprocessing, feature extraction, normalization, embedding, and calculating a similarity score, all in real-time, to make an authentication decision of accepting or rejecting. The effectiveness of the proposed system is recognized mainly when a user tries to authorize a transaction using their voice, and the system correctly identifies the speaker or rejects the speaker based on the stored reference. To achieve this goal, advanced mechanisms like adding variety in the dataset or data augmentation are incorporated in the system. Along with this, through the noise handling mechanism, distorted data has been processed to maintain the robustness of the system. Therefore, considering diverse conditions like noisy environments, various emotional conditions, and different accents, these mechanisms make the system robust and reliable for real-life practices. Besides performing functional requirements, the system equally addressed nonfunctional requirements. Performance efficiency is one of the main non-functional aspects that guarantees the system's quick response under substantial amounts of data processing, which ensures that the user experiences minimum waiting time. Accuracy, another non-functional requirement, ensures that the system can detect the actual speaker, differentiating between actual and fraudulent user voices. This builds the system as a trustworthy and efficient digital transaction system in a practical application that encourages people to be involved in digital transactions. Scalability is considered one of the key aspects of the non-functional specification that handles the growing number of users and larger datasets without degrading the performance of the system. Reliability of the system ensures it can provide correct results even if the voice is misrepresented by noisy surroundings or various emotional conditions. This makes the system more dependable to the users for financial transactions like mobile banking or a mobile payment system. Lastly,

Usability focuses on providing a user-friendly experience and making the system easily accessible to the user. Together, all these specifications provide a precise and coherent framework that upholds the validity of the system while promoting its adaptivity and extending its sustainability.

3.2 Societal Impact

Besides the technical implementation, the proposed system is expected to generate a large spectrum of Social welfare, offering many positive impacts on society. Introducing Voice-based authentication, the system not only addressed security in financial transactions, but also introduced easy accessibility and a trustworthy digital transaction system to the community where most of the traditional systems, like PIN and signature, are unsafe nowadays for transferring money and resources. So, by producing reliable and secure output, the system guarantees safety and security in financial transactions, and also improves the lives of the people in society. This is particularly beneficial and valuable for underdeveloped countries, as a significant percentage of the population of those countries has lower literacy rate, and is unable to access many advanced technologies, due to their literacy barrier and lack of technical knowledge. But this system assures that it is not only accessible to the privileged group, but also widely accessible and easily usable to a diverse population who have less technical knowledge. Thus, this system promotes digital equality and brings social well-being, and makes citizens gain confidence in involving themselves more in the financial transaction system. When it comes to safety and personal well-being, the system lowers the probability of the risk that may arise from errors, mistakes, and scams. High accuracy and reliability of the system minimize the chance of misclassification and unauthorized access, which ensures the assets of the citizens of the society are preserved. This reliability also builds confidence in users to be involved in digital banking and the mobile payment system. Thus, individuals can safeguard their personal resources and money with easy access to digital transactions without worrying about fraud or scams, which also brings mental peace. Additionally, the system carries high ethical and legal values. It ensures data is carefully used and maintained ethically to safeguard people's privacy. As the system uses personal voice data, the technologies are being used keeping in mind that the data will not be misused in any other manner. It reflects that the data is handled carefully and securely while upholding the rights of its users. By doing this, it ensures trustworthiness among the users as well as servers as a model of future technologies in the sensitive financial context. From the perspective of cultural sensitivity, the system is better suited for local users of the Bangla-speaking region, as it supports Bangla voice data in contrast to technologies that are developed for a Western audience. Thus, it prioritizes local communities and makes it usable and easily accessible to the local people rather than the privileged Western community. By addressing cultural and linguistic needs, the system guarantees wider acceptance and is embraced by a wider range of users in their everyday use.

3.3 Environmental Impact

The main purpose of the proposed system is to increase security in financial transactions, but it offers benefits to the planet by creating positive effects on it. As digital technology goes worldwide, it is important to develop systems that are not only technically efficient but also environmentally friendly. By reducing energy, physical wastage and making better use of existing technologies, it supports the objective of making the proposed system eco-friendly. Fundamentally, this system reduces printed paperwork, physical storage, documents, and physical storages, which is linked to the traditional banking systems. But by switching to a digital voice-based authentication system minimizes the environmental costs. Technically, the system is designed to increase efficiency in digital transactions. To prevent processing data repetitively, it uses methods like data segmentation, noise augmentation, and normalized feature extraction. Accomplishing the variety and expansion of the dataset are achieved through a smarter process for avoiding copying data again and again, which eventually saves memory and energy. This indicates the system can manage big tasks without much power and energy consumption, whereas many AI-based model consumes a lot of power. When it comes to implementation, the system can be deployed on the same devices, such as mobile phones and online banking apps, which are already familiar to the users. Thus, the system minimizes the electronic wastage by preventing the purchase of new electronic devices for using the system. By expanding the life of the existing device, the system improves the sustainability of the digital ecosystem. Additionally, the system encourages greener practices, which are environmentally friendly. For example, in the traditional banking system, users need to go to the bank physically for verifications, but in the digital verification system, they don't need to go to the bank physically, which saves valuable time of the users. The system also lowers the transportation-related emissions, which eventually reduces carbon impact on the environment. Thus, the system demonstrates it can be equally technologically viable as well as ecologically advantageous.

3.4 Ethical Issues

Ethical considerations play a very important role in any financial authentication system to make the system fair, reliable, and safe for all users. Keeping in mind these values, the proposed system is designed in such a way that it can be implemented with confidence in real-world applications. Privacy is one of the key concerns as voice data is unique and varies from person to person; misusing or sharing personal voice data without their consent can bring major harm. Considering the sensitivity of the data, the system is designed to gain confidence among the users and involve them fearlessly in the digital financial services. Due to the system's strong security, users gain trust and can confidently interact with digital financial transactions. Another important part of the system is fairness. Many traditional technologies are designed for a certain group of people without considering the linguistically and culturally diverse community. This system assures that culturally diverse people who have different accents are equally able to interact with the system, avoiding bias. Thus, people of all cultural backgrounds are treated equally and fearlessly accessing the digital financial services.

Since academic and professional responsibilities are the foundation of reliable and

responsible research, these obligations are taken seriously into consideration. It indicates that during the development of the system, respect for intellectual property has been maintained. The ideas, methodologies that are borrowed from others' work have been acknowledged properly, and appropriate citation of all the sources has been added to show respect towards the original author. It clearly demonstrates that the system has maintained proper academic integrity. Additionally, the system is developed with the integration of honesty and reliability as guiding principles. This indicates that the outcome is not misinterpreted or exaggerated rather shows the actual performance and limitations of the system, which ensures the trustworthiness both in academic and practical applications. By upholding the standards, the system proves itself both as a credible academic resource and a trustworthy financial authentication.

3.5 Project Management Plan

A well-structured Project management plan is essential to accomplish the successful development of the proposed system. To achieve the goal, the project has been divided into several different stages. Each stage is designed to complete its own objective. During the first stage, the requirements analysis of the needs of financial authentication and identifying the existing challenges of the traditional transaction systems has been executed, and the next stage, system analysis, includes planning the architecture, finding ways to preprocess the data, and choosing the algorithm. Then it comes to the actual system building, where data preprocessing, feature extraction, appropriate model training, and evaluation have been integrated. In the implementation stage, after preprocessing data, several models have been trained to identify age, gender, regional dialect, AI, and similarity between users. Through the Siamese model similarity between two users has been compared using a similarity score. An AI-generated fraud voice detection technique has been developed in the system using a probability model. Alongside, identifying the age and gender of the user mechanism is incorporated in the system through the Age and Gender model. After the implementation of the system, the accuracy and reliability of the system are evaluated by testing it with practical scenarios through the evaluation stage. Throughout the entire process, the tasks were maintained with regular monitoring, available resources, and time. This structural approach helped to handle obstacle management and develop the desired system within the allocated time and resources.

3.6 Risk Management

Risk management of any system development project makes it easier to find the potential issues early and determine effective solutions in advance. To avoid possible problems, several risks were carefully considered throughout the development of the proposed system. Poor quality data is one of the main risks, as voice data can be noisy or unclear. Preprocessing and noise handling mechanisms have been incorporated during the development of the system to increase the system's performance. Bias in the model is considered another major risk, which can reduce the system's accuracy and fairness. It mostly occurs when any system is not trained with a diverse dataset where culturally and linguistically different people are ignored. But

this system represents all kinds of Bangali speakers who have different accents and cultural values. Efficient models and algorithms are implemented to avoid various technological risks, which ensure optimization and accuracy of the output. Finally, time and resource risks were managed with proper division of the system and careful task management. By addressing these risks earlier and taking efficient action, the chances of obtaining an accurate result from the system have been increased. Besides, it helped the system to enhance its robustness and confidently perform real-world challenging tasks.

3.7 Economic Analysis

Economic analysis is considered an important part of the system development project as it measures the cost and benefit of the project, identifying whether the project is economically feasible or not. Many systems possess great technical usability but have lower economic usefulness because of their high expenses in the development stages. Then, despite having technical viability, these systems can not be implemented widely due to their lower cost-effectiveness. That's why the study of economic analysis of the system has been carried out to check whether the proposed system is applicable or not to the real-world digital transactions. The analysis considered both indirect and direct costs involved with the stages of the development of the system, starting from data collection, preprocessing, model training, and testing. Time and effort from the developers have also been considered, as planning, building, improving, and testing the system requires many working hours. The total cost of the system is made up of all these elements. Apart from significant working hours and efforts of the developers, the proposed system required very minimal expenses for accessing necessary resources, as during the development, maintaining cost effectiveness was the key priority. On the other hand, the system possesses long-term benefits which outweigh its initial costs. Once the system is implemented, it reduces huge manual working hours, costs of the physical documents, and transportation costs for the users, whereas traditional transaction systems require all these, making the system costly and less affordable. Therefore, analysing the cost and benefit of the system, the system demonstrates that the solution is not only technologically viable but also economically beneficial for real-world applications.

Chapter 4

Methodology

4.1 Dataset Collection

The primary dataset for this study, **"Mozilla Commonvoice voice corpus 20.0"** [43] is a dataset composed of short clips of Bengali speech in mp3 format from a magnitude of sources of different gender classes ['male_masculine', 'female_feminine'] and age range ['teens', 'twenties', 'thirties', 'forties', 'fifties']. It also has a train, test listing with classification. The proposed models run following these listings. The prominent age class in the dataset is the twenties class, and male is the prominent gender. The train set contains a total of 21375 instances and the test set has 9356. The number of null values in the dataset is not negligible and there is class imbalance, these problems were mitigated in the data-preprocessing phase. There was only one instance of a voice of the fifties age class so it is removed.

The dataset comprises several key columns, including client_id, path, sentence_id, sentence, sentence_domain, up_votes, down_votes, age, gender, accents, variant, locale, and segment, which collectively define the structure and metadata of each audio entry. The initial samples of these columns from both the training and testing datasets (similiar as training) are illustrated in Figure 4.1.

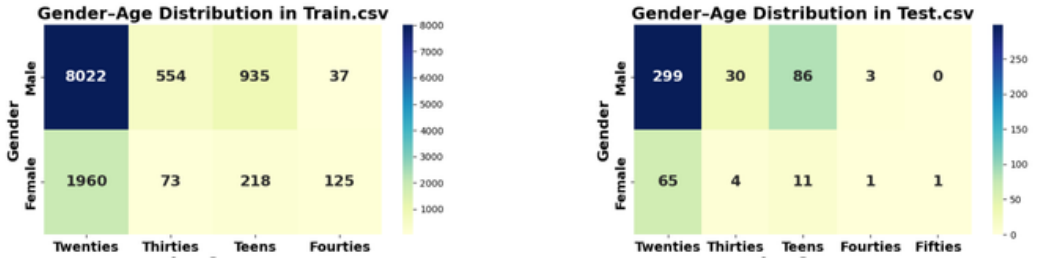
client_id	path	sentence_id	sentence	sentence_...	up_votes	down_votes	age	gender	accents
22fcb186dbdc5da704cb2d3477bef20af74db5ea7eab5317ee5ea82dc1243c32f53ba786f25810b25d46ddc958d3d71701bd...	common_voice_bn_31514844.mp3	2304e78a1465e17ed36d416c76d6cb2182a1e6aafdc6954c52be8b7ea341334f	তিনি তার সমস্তকালের খুব স্বল্পসংখ্যক ক্রিকেটারদের একজন যিনিবে ইংল্যান্ডের বিপক্ষে ভারত দলের সদস্যরূপে...		6	0			
22fcb186dbdc5da704cb2d3477bef20af74db5ea7eab5317ee5ea82dc1243c32f53ba786f25810b25d46ddc958d3d71701bd...	common_voice_bn_31515597.mp3	22b85d6ba6160641264c1579eb48d7d1ad2a157b8cc366f8cfdace959f8d0e18	যখন একাধিক শব্দ একত্রিত হয়ে একটি সর্বনাম তৈরি করে, তখন তাকে যৌগিক সর্বনাম বলে।		10	2			
22fcb186dbdc5da704cb2d3477bef20af74db5ea7eab5317ee5ea82dc1243c32f53ba786f25810b25d46ddc958d3d71701bd...	common_voice_bn_31540854.mp3	3626d15ddc02bee80826e9158c6852b43323a1e3a7a1f62896ac584840644992	এটি জয়পুরহাট শহরের পাঁচবিবি রোডে অবস্থিত।		2	0			
22fcb186dbdc5da704cb2d3477bef20af74db5ea7eab5317ee5ea82dc1243c32f53ba786f25810b25d46ddc958d3d71701bd...	common_voice_bn_31541165.mp3	371b2047eb46976023c292fb84c35d6f5c2fd50150feea0a2e018b3e60885d19	মামুনের প্রতি তার অগাধ ভক্তি থাকা স্বত্বেও পিতাকে তার রচিত গ্রন্থের সর্বত্রে মামুন প্রীতি প্রদর্শন করে...		17	7			
23fe036a8f971e3b68fa2a9fbe2872e19736cb8c444cbbf3234d74cb0e3992dff31bd2165c2fc27582725d6ff6	common_voice_bn_31590843.mp3	6f654e2882a244a6011cdb3aa5b37d156710e46accb728b21acb0c9c874dbf83	কক্ষের উপরে উচ্চতার একটি গম্বুজ রয়েছে।		2	1			

Figure 4.1: First 5 samples of the train data.

The Bangla common voice **'Mozilla Common Voice'** dataset for gender and age classifications contains more than 16000 Bangla audio samples which help the model to train well and the data set consists of real-world audio data with noise and diverse acccents, which helps to make the model robust and increase the model accuracy for voice signature recognition.

4.1.1 Dataset Distribution:

The heatmap graphs in Figure 4.2a and 4.2b illustrates how the data in the training and test datasets is dispersed across different gender and age groups. The color intensity indicates the sample number in any gender-age class. Darker color denotes higher samples in any gender-age class. In the Train.csv heatmap, the age group 'twenties' dominates with the most representations, where the male has 8022 samples and the female has 1960 samples. 'teens' group has a notable male representation with 935 samples. The 'Forties' and 'thirties' groups have very insignificant female samples whereas the 'thirties' group has 73 samples and the 'forties' group has 125 samples. In Test.cvs heatmap , 'twenties' group has the highest number of representation with 299 male sample and 65 female samples . The other groups 'teens' and 'thirties' have fewer male and female samples with 86 male, 11 female samples and 30 male, 4 female samples correspondingly.the rest of the groups 'forties' and 'fifties' have negligible male and female samples. From these heatmaps male sample domination is noticeable in most of the age group. Figure 4.2a and 4.2b summarize the gender–age distribution in the training and test datasets, respectively, highlighting that the majority of samples belong to males in their twenties.



(a) Train.csv Gender–Age Distribution (b) Test.csv Gender–Age Distribution

Figure 4.2: Side-by-side comparison of gender–age distributions in the training and test datasets.

4.1.2 Data Preprocessing for gender and age classification model

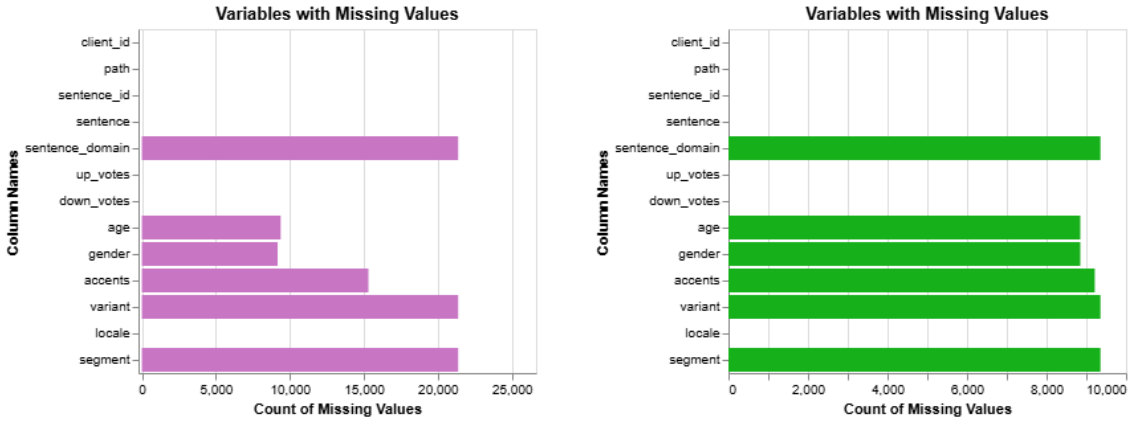
4.1.2.1 Data Pruning:

Data pruning removes incomplete and inconsistent entries from the dataset. Rows with missing values for gender or age are deleted. Rows with invalid file paths or empty labels are also excluded, since these samples cannot be used for training. Gender and age labels are converted into lowercase to maintain a uniform format across all samples. For age, raw values are grouped into fixed categories based on decades. The groups include less than 20, twenties, thirties and forties. Samples

that do not fall into these groups are removed. This ensures that all age information follows the same category format. For the age group 50-59 is dropped as it is present is Test.csv but missing in Train.csv.

After pruning, the dataset contains only entries with complete and valid information. The gender-age distribution after pruning is shown in Figure 4.3, which compares the missing value distributions between the training and test datasets.

The corresponding gender–age distributions for the training and test datasets are summarized in Table 4.1 and Table 4.2, respectively. The training data (Table 4.1) shows a significantly higher representation of participants aged 20–29, while the test set (Table 4.2) maintains a similar pattern but with fewer total entries.



(a) Missing values in Train.csv

(b) Missing values in Test.csv

Figure 4.3: Comparison of missing value distributions between Train and Test datasets

Gender	**Age Group**				
	40-49	**30-39**	**20-29**	**20-19**	**<20**
Male	37	554	8022	935	0
Female	125	73	1960	218	0

Table 4.1: Train.csv Gender–Age Distribution

Gender	**Age Group**				
	50-59(X)	**40-49**	**30-39**	**20-29**	**>20**
Male	0	3	30	299	86
Female	1	1	4	65	11

Table 4.2: Test.csv Gender-Age Distribution

4.1.2.2 Audio Preprocessing:

Raw audio files are loaded and normalized using the librosa library. All audio files are resampled to a Standardized Sampling Rate rate of 16,000 Hz to ensure uniformity

across the dataset. Stereo audio files are converted to mono to simplify analysis to reduce computational complexity. The Audio clips are then standardized to a fixed duration of three seconds, clips longer than three seconds are truncated, while shorter clips are padded with zeros.

4.1.2.3 Spectrogram:

Frequency over time representation is depicted by this part where audio data is converted to determine unique characteristics from analyzing frequencies. To determine variation in the frequency pattern of male and female speakers, average male and female spectrograms are measured. As raw audio data contains unnecessary elements such as noise and silence; preprocessing is necessary for system optimization ensuring only cleaned waveform is converted into a spectrogram for feature extraction.
Silence removal is implemented via amplitude threshold($|signal|<0.01 \rightarrow 0$) to eliminate non-vocal segments, followed by aggressive noise suppression through elevated threshold filtering (0.05 cutoff) to attenuate low-energy artifacts to prepare the data for feature extraction
Most audio samples in the dataset are clean, the small noises that were present however are percussions, so it is decided to remove them. Whispers and less prominent noises (signal threshold) are also removed. Audio voice and harmonics are also removed and represented them in the following spectrogram. The initial audio waveform, showing the separation of voice and harmonic components before preprocessing, is illustrated in Figure 4.4.

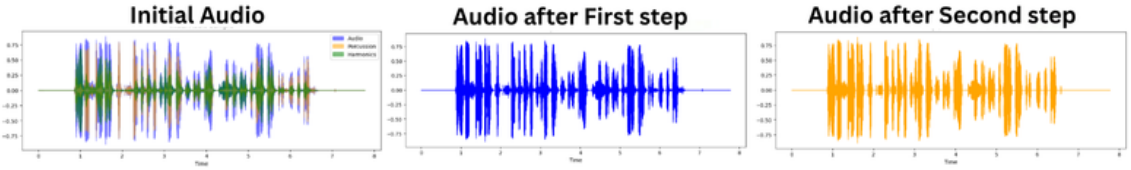


Figure 4.4: The initial audio waveform before data preprocessing

Average Mel Spectrograms:

- **Gender-based Spectrogram:** These spectrograms illustrate the average frequency over time for males and females, highlighting differences in voice frequency. This information is useful for developing gender-sensitive voice signature recognition models. The results are shown in Figure 4.5.
- **Age-based Spectrograms:** These spectrograms depict frequency patterns across different age groups, allowing the system to capture voice variations related to age. The corresponding spectrograms are presented in Figure 4.6.

Due to diverse data in the gender age group with over 10000 data samples The Mozilla Common Voice dataset plays a vital role in developing an accurate model for Bangla voice recognition tasks. Though throughout the whole dataset male samples dominates, and biases are noticed in gender groups where 'twenties' age group has the highest samples, by pitch sifting , frequency masking and other techniques biasness has been reduced through data balancing which ultimately helped achieve a fair model for voice recognition authentication.

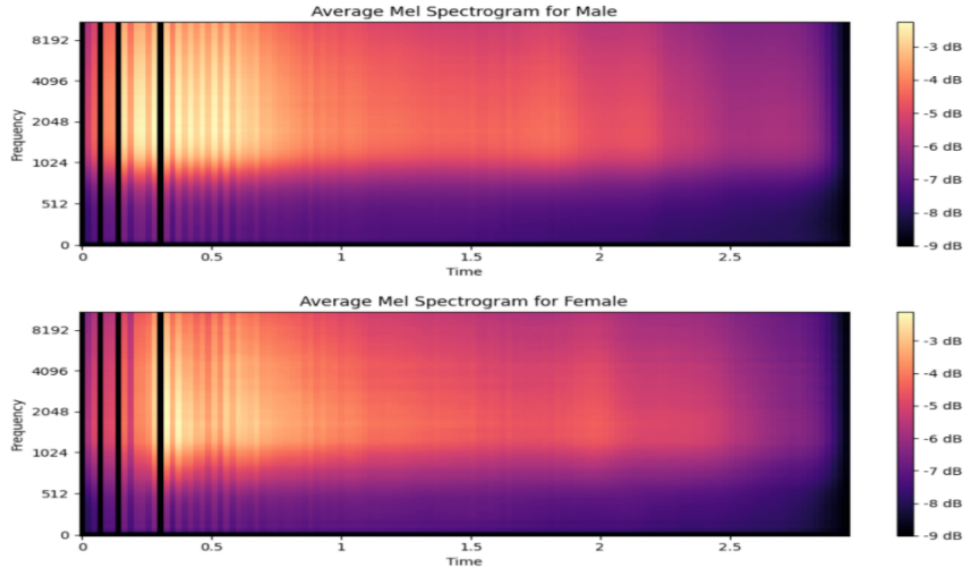


Figure 4.5: Gender-based spectrogram (all classes) after data preprocessing.

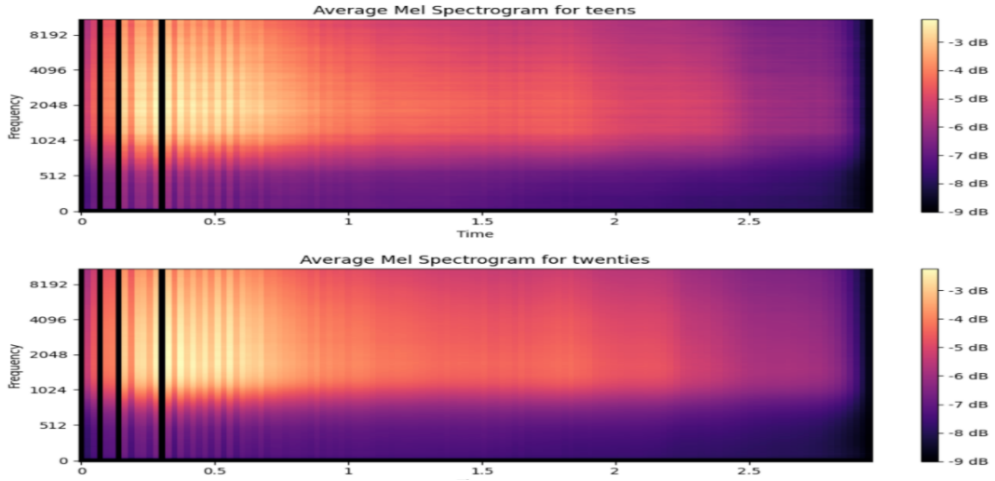


Figure 4.6: Age-based spectrogram (all classes) after data preprocessing.

4.1.3 Balancing Dataset Through Up-Sampling Implementation:

In the dataset, there exists gender and age group imbalances, which might cause models to show preference towards major classes thus decreasing both accuracy and fairness of the system. To solve this issue, dataset up-sampling is applied to balance the dataset. Minority classes are upsampled by sample generation using randomized selection with replacement and feature-space augmentation, while majority classes undergoes down sampling to mitigate bias capping maximum samples per category (5000 per age group and around 10000 per gender group) to prevent overfitting.
The voice samples in the dataset exhibits unbalanced gender distribution because of which predictions exhibits a bias toward male samples in the dataset. The code implements balance_gender_classes() as a function that expands the minority gender population through sample duplication and introduces data augmentation methods to maintain data diversity beyond mere duplication. The data augmentation

techniques includes the process of adding random noise and using both frequency masking (hiding random frequency bands) and time masking (hiding random time intervals) techniques.

Similarly, the age-group balancing was necessary as the samples from some age groups, example, teens,forties, age ranges are much fewer than those from age groups like twenties and thirties. The balance_age_samples() function performed data oversampling for rare age groups including teens and forties to obtain balanced dataset. All gender balancing augmentation processes such as random noise addition and frequency and time masking technologies are also used here. In addition to that, time-stretching (to simulate slower to faster speech) and pitch-shifting techniques are specifically applied to simulate natural speech variations which occur in different age groups during the artificial sample creation process. The developed up-sampling strategies improve the dataset's representativeness, enabling better generalization, fairness, and robustness in age classification outcomes, as illustrated in Table 4.3, which presents the balanced gender and age group distribution after applying the up-sampling techniques.

Gender	**Age Group**			
	40-49	**30-39**	**20-29**	**<20**
Male	2178	2178	2178	2178
Female	2822	2822	2822	2822

Table 4.3: Gender and Age Group Distribution

4.1.4 Feature Extraction

To extract features from raw audio into meaningful representation the torch-audio library is used, the raw audio waveform are converted into a spectrogram with 128 Mel bands for frequency representation. This time domain audio signal is run through Logarithmic scaling (log10) for ease of representation by reducing the dynamic range. Now multiplying it by 10 would give out true decibel scale 10*torch.log10(mel_spec + 1e-9) but, it is ignored as the data is being for training purposes. This extracted Mel-Spectrogram serves as a compact representation of the audio signal's time-frequency (features), making it suitable for classification tasks including binary gender classification and non binary age classification.

4.2 Dataset for AI Detection

For the AI voice detection task, the Mozilla Common Voice dataset [43] is employed as the primary source of authentic human speech. From this corpus, an AI-generated counterpart is synthesized to create artificial speech samples using NVIDIA Riva [44] Toolkit. This process ensured a balanced representation between human and synthetic voices while maintaining diversity in accent, tone and recording conditions. The resulting dataset provides a standardized and high-quality foundation for training a deep neural network capable of distinguishing human voices from AI-generated speech.

4.2.1 Dataset Collection and Validation

The dataset metadata includes information for each recording, such as the file path, the label indicating whether the voice was human or AI-generated, and the original source file to track speaker identity. To ensure the integrity of the training data, all entries with missing or invalid file paths are removed, and all labels are verified to belong to one of the two classes: 'human' or 'AI'. Each label is mapped to a binary index (0 for human, 1 for AI) to standardize the representation for model training.

Audio files are then loaded and standardized. Each recording is resampled to 16 kHz and converted to mono to ensure uniform sampling rates and channel configuration. To maintain temporal consistency, all clips are either trimmed or zero-padded to five seconds, producing waveforms of equal length across the dataset. This uniformity allowed for consistent batch processing and feature extraction, critical for subsequent neural network training.

4.2.2 Feature Extraction

Acoustic features are extracted from each waveform as 128-dimensional log-Mel spectrograms. These spectrograms preserve both spectral and temporal information of the speech signal while reducing the dimensionality for computational efficiency. The features were log-compressed to stabilize the dynamic range and normalized to zero mean and unit variance, ensuring consistent input statistics for the network. The resulting spectrograms provided a structured, high-resolution representation of the speech, suitable for convolutional and recurrent neural processing.

4.2.3 Data Augmentation

To improve model generalization and reduce overfitting, both human and AI-generated recordings are augmented. Human recordings are enriched with realistic environmental noise, whereas AI-generated samples are diversified using a wide variety of transformations, including pitch shifting, time stretching, reverb, distortion, robotic modulation and formant shifts. These augmentations simulated variations in recording conditions, vocal characteristics and synthetic modifications. In total, the AI class included hundreds of augmented variations to ensure sufficient diversity. This augmentation strategy ensured that the model encountered a broad range of speech patterns across both classes.

4.2.4 Class Balancing

The original dataset exhibits a natural imbalance, with human recordings generally more numerous than AI-generated samples. To mitigate potential bias during training, the AI class is up sampled using augmented and synthetic variations, while the human class was down sampled to a comparable number of samples. The resulting balanced dataset provided equal representation of both classes, enabling the model to learn without bias toward the majority category.

4.2.5 Dataset Statistics

The final curated dataset consisted of 128,250 samples, split equally between human and AI-generated voices. Each class included 64,125 samples, with five-second waveforms standardized across the dataset. Table 4.4 summarizes the key statistics of the dataset. The dataset also contained 1,353 unique voices, ensuring speaker diversity across both classes.

Class	**Original / Noisy**	**Augmented**	**Total**
Human	64,125	—	64,125
AI-generated	—	64,125	64,125
Total	—	—	128,250

Table 4.4: Distribution of samples in the AI detection dataset, showing original, augmented, and synthetic recordings for human and AI-generated classes.

Overall, the dataset preparation ensured that all audio samples are standardized, balanced and sufficiently varied to capture real-world variability. The combination of careful curation, feature extraction, data augmentation and class balancing provided a excellent foundation for training the AI detection model. This curated dataset allowed the model to learn meaningful acoustic patterns for distinguishing between human and AI-generated voices while minimizing the risk of overfitting to specific speakers or synthetic artifacts.

4.3 Dataset for Regional Dialect

The vocal characteristics of individuals in Bangladesh vary significantly across districts. This variation extends beyond differences in gender or age, reflecting unique local dialects, rhythm, pronunciation patterns, and linguistic traits specific to each region. In voice-based recognition systems, it is not sufficient to verify a speaker's identity; recognizing their regional characteristics can provide additional insight into linguistic patterns and speaker profiling.

This study aims to develop a custom deep learning model capable of automatically identifying region-specific dialects from raw audio recordings. By leveraging an end-to-end architecture that combines local feature extraction and temporal modeling, the project transforms raw voice recordings into meaningful district-level predictions, offering a novel perspective on Bangladeshi voice recognition.

The foundation of any speech-based dialect classification system lies in the quality and structure of the dataset. In this work, the publicly available another dataset 'RegSpeech12' [45], which provides recordings annotated with speaker information and corresponding district-level labels. Since the raw dataset is not directly suitable for training deep neural networks, several stages of preparation and refinement were applied. These included metadata validation, waveform preprocessing, feature normalization, data augmentation, and balancing of class distributions. Each stage ensured that the final dataset was consistent, representative, and robust against variability, thereby enabling reliable training of the proposed Bi-LSTM-based model.

4.3.1 Dataset Description

The dataset consists of speech recordings in Bangla Language collected from multiple districts of Bangladesh. Each entry includes an audio file and associated metadata specifying the district of origin. The dataset provides natural conversational speech, making it suitable for dialect identification tasks. A total of 13,342 recordings are included, distributed across ten districts, though the representation of districts was uneven. Table 4.5 summarizes the raw distribution of the dataset before preprocessing.

District	Number of Samples
Sylhet	2,903
Kishoreganj	1,638
Narail	1,488
Chittagong	1,406
Narsingdi	1,098
Sandwip	1,049
Rangpur	1,037
Tangail	987
Habiganj	940
Barishal	796

Table 4.5: District-wise distribution of samples in the dataset.

Observation from the dataset: Districts such as Sylhet and Kishoreganj are more heavily represented, while Barishal, Tangail and Habiganj have fewer samples. This imbalance, if unaddressed, could bias the classifier toward majority districts. A graphical representation of this imbalance is shown in Figure 4.5.

4.3.2 Metadata Validation

Prior to waveform preprocessing, the metadata is carefully examined. Records with missing or invalid file paths are discarded, and district labels are standardized to ensure uniformity. Only entries with valid labels corresponding to the predefined district list are retained. After this filtering step, the dataset size remained at 13,342 samples, ensuring data integrity and eliminating noisy entries.

4.3.3 Preprocessing of Audio Waveforms

To prepare the dataset for neural network training, each audio recording was processed as follows:

- Resampled to 16 kHz and converted to mono.
- Trimmed or zero-padded to a uniform duration of 5 seconds.

This preprocessing ensured that every input to the model has the same temporal dimension, avoiding mismatches during batch training.

4.3.4 Feature Preparation and Normalization

While the proposed model learns acoustic features directly from raw waveforms using convolutional layers, normalization is applied to stabilize training. Each waveform is mean-normalized and divided by its standard deviation, resulting in features with zero mean and unit variance. This step improved statistical consistency and controlled for amplitude variations across recordings.

4.3.5 Data Augmentation

To improve model results and prevent overfitting, data augmentation is applied on waveforms:

- **Frequency Masking:** Randomly masks a consecutive range of frequency channels in the spectrogram, simulating variations in pitch, timbre and recording conditions. This helps the model become invariant to slight changes in speaker voice characteristics and background noise.
- **Time Masking:** Randomly masks consecutive time frames in the spectrogram, mimicking pauses, dropped frames or irregular speech patterns. This encourages the model to learn temporal features more robustly, reducing sensitivity to small timing variations.

Augmentation is applied mainly to underrepresented districts to improve diversity and balance.

4.3.6 Balancing the Dataset

The dataset is naturally imbalanced, with some districts contributing many more samples than others. To mitigate bias, oversampling with augmentation is applied to minority districts such as Barishal, Habiganj and Tangail. After augmentation, the dataset achieved a more balanced distribution across districts while maintaining a total of 13,342 samples.

4.3.7 Final Curated Dataset

Following validation, preprocessing, normalization, augmentation and balancing, the final dataset consisted of 13,342 recordings, each standardized to 5 seconds. The dataset is then shuffled to remove ordering bias and divided into training and test splits in the ratio 72:25, resulting in:

- **Training samples:** 10,006
- **Testing samples:** 3,336

This final curated dataset is both statistically balanced and acoustically consistent, providing a strong foundation for training the regional dialect classification model.

4.4 Data Preprocessing for Siamese Model

Preprocessing is a step-by-step process to prepare raw audio for model training. It began with dataset preparation and continued with feature extraction and normalization.

4.4.1 Dataset Preparation

Audio files are loaded from the `clips` directory using `Train.tsv` and `Test.tsv` and merged. Rows with missing or invalid metadata such as path, gender, age or clients with fewer than two recordings are removed. To reduce imbalance, clients with more than fourteen recordings are down sampled by keeping ten to fifteen of their longest recordings.

The dataset is then divided into training and testing sets on a per-client basis, ensuring that both sets contained the same clients, but their recordings did not overlap. Smaller clients contributed one or two test samples, while larger clients contributed about 20% of their recordings.

Each recording is standardized to five seconds. Longer files are split into overlapping five second clips using a stride of 1.7 seconds, while shorter ones are padded. Noise augmentation is applied through `noise_generator.py`, which produced real and synthetic noise in four categories: urban, crowd, nature and indoor. Noise is mixed with speech at random SNR levels.

Training data used a 90% probability of augmentation with SNRs between 10–20 dB, while testing used a 70% probability with SNRs between 5–25 dB and a separate random seed. Volume scaling is also applied. Each training client is increased to 70–80 samples through augmentation, while test clients are expanded by adding 2–4 augmented versions per recording. The output is a set of five-second clips in the `clips/` directory with updated `train.tsv` and `test.tsv` files.

4.4.2 Feature Extraction

Each recording was resampled to 16 kHz, normalized and converted into frame-level features including:

- **MFCCs with deltas:** Shows the main shape of the speech spectrum and how it changes over time.
- **Pitch:** Gives the base frequency of the voice, useful for telling speakers apart.
- **Formants (F1–F3):** Shows the main resonances of the vocal tract that shape different speech sounds.
- **Jitter and Shimmer:** Measures small changes in pitch and loudness, linked to voice stability.
- **Harmonic-to-noise ratio (HNR):** Shows how clear the voice is by comparing harmonic sound to noise.

- **Spectral centroid, Bandwidth, Roll-off, Flatness and Contrast:** Describe how sound energy is spread across different frequencies.
- **Root mean square energy (RMSE):** Measures the strength or loudness of the speech signal.
- **Zero-crossing rate (ZCR):** Counts how often the signal crosses zero, helping to separate voiced and unvoiced sounds.
- **Chroma and tonnetz:** Capture tonal and harmonic patterns in the speech.
- **Voice activity probability:** Detects which parts of the signal contain speech and ignores silence.

Features are stored in a fixed length by padding. Utterance level features are created by summarizing frame level values using mean and standard deviation. A merged representation is built by combining frame-level and utterance-level features into multi channel arrays. Features are then saved in HDF5 (`.h5`) format in three sets: dynamic (time-series), static (summarized descriptors) and merged. Separate files are generated for training and testing under `features_h5/train/` and `features_h5/test/`.

4.4.3 Normalization

Normalization is applied to dynamic frame level features before pooling, as implemented in `normalizer.ipynb`. Each frame was scaled using the normalization formula shown in Equation 4.1.

$$\text{features}_{\text{normalized}} = \frac{\text{features} - \mu}{\sigma} \tag{4.1}$$

where μ and σ are the mean and standard deviation of each feature channel. This brought all recordings to the same scale.

For static features, normalization is not applied, since they were computed only after aggregating frame level values into overall mean and standard deviation descriptors.

4.4.4 Pooling and Model-Specific Handling

For the **Siamese model**, pooling is performed using a Voice Activity Detection (VAD) mask generated by Silero VAD (`get_speech_timestamps`). Only frames containing speech sare sed for pooling and silent intervals are excluded. This ensured that silence does not affect the averaged embeddings, which is especially important for recordings with long pauses, such as Bengali speech.

Both normalized frame level features and their pooled embeddings are stored in HDF5 format under `features_h5/train/` and `features_h5/test/`. The Siamese model used the pooled embeddings for training and inference. Keeping both frame-level and pooled versions allowed future experiments without repeating feature extraction.

The system stored both normalized frame-level features and their pooled versions in HDF5 format under `features_h5/train/` and `features_h5/test/`. Sequential models such as BiLSTM used the normalized frame-level sequences directly, while non-sequential models such as the Siamese model used the pooled embeddings. Keeping both forms allowed future experiments without repeating feature extraction.

4.5 Proposed Models

In this section the different models designed and implemented in this research are discussed. Each model is introduced with its architecture and role in addressing the classification tasks.

4.5.1 Modified Bi-LSTM for gender and age classification

The Bidirectional Long Short-Term Memory (Bi-LSTM) processes speech sequences in both forward and backward directions, so it can use information from both past and future frames. In this work, **Modified Bi-LSTM** is used to classify both gender and age from speech. First, the speech features are passed through a shared encoder, then split into two parts: one branch with a three-layer Bi-LSTM and attention for gender (male/female), and another branch with a four-layer Bi-LSTM with multi-head attention and skip connections for age groups (teens, twenties, thirties, forties). These changes make the model suitable for handling both gender and age classification within the same design.

The customized Bidirectional Long Short-Term Memory (BiLSTM) model employed is designed in a way that it will handle both classification tasks (for both gender and age) simultaneously while sharing a common feature extraction pipeline. This section provides a detailed description of the Modified BiLSTM model architectures. The model is also modified to give improved accuracy through skip connection and condensation and more complex dense layers for age classification.

4.5.1.1 Modified Bi-LSTM Model Architecture:

The proposed Modified Bi-LSTM model consists of three main components: A shared encoder, A gender-specific classification segment and An age-specific classification segment.

The shared encoder is responsible for extracting high-level feature representations from the input audio features (mel_spectrogram). First the encoder performs linear transformation with a fully connected layer and the input features are encoded to the desired lstm(hidden layer) unit size. After that, Layer Normalization is done. The output goes through ReLU activation function to introduce non-linearity and finally a 40% dropout rate is selected. This shared encoder serves both gender and age classification tasks.

The gender classification pathway is simple as it is a binary classification. It starts off with a three-layer bidirectional LSTM network(hidden layers), which captures tem-

poral patterns in the shared (age,gender) feature representations from the encoder. Bidirectional processing lets the model consider both past and future context within the audio sequences. An attention mechanism (Residual Attention) is introduced that enhances the model's focus on key temporal features of the LSTM output, this somewhat improves classification accuracy for the test dataset. Finally for classification, a simple dense layer that takes input from both the forward and backward LSTM layers and the provided output is normalized, passed through ReLU activation function and regularization is done on it. A second dense layer then predicts the audio to either be a male or a female voice.

The age-specific pathway is designed to handle the more complex task of age classification of which we have four category ranges: teens, twenties, thirties and forties. This pipeline starts with a four-layer bidirectional LSTM network that captures subtle changes in temporal patterns, this increase in complexity is required for distinguishing between the multiple age groups.

Multi-head attention is applied using nn.MultiheadAttention, it processes the LSTM output and focuses on salient temporal features. Each attention head learns distinct patterns or relationships within the feature sequences. Using a skip connector the raw encoder feature tensor is concatenated to this and the max pooling is done. Now for classification, this processed feature-set is passed through three dense layers and the age range of the sample is predicted. The outputs of each intermediate layer are normalized; as illustrated in Figure 4.7, which depicts the workflow of the proposed Modified BiLSTM model architecture.

4.5.1.2 Training of the Gender Classification Model using Modified BiLSTM

The training of the gender classification branch begins with batches of speech samples, each represented as a 128-dimensional acoustic feature vector. These input features first pass through the shared encoder, where they are projected into a 256-dimensional latent representation using a linear layer. The output is normalized with Layer Normalization, passed through a ReLU activation, and processed with dropout (0.4) to reduce overfitting.

The encoded features then enter the gender branch. Here, they are processed by a three-layer bidirectional LSTM with hidden size 256. This allows the model to capture information from both past and future time steps in the speech sequence. Between LSTM layers, dropout (0.4) is applied to limit overfitting.

The output of the BiLSTM is refined using a residual attention mechanism with dimensionality 512. This mechanism learns to focus on more important time steps while keeping the original information through residual connections. After attention, mean pooling is applied across the time dimension, producing a fixed-length vector where each frame contributes equally.

This vector is passed through a two-layer fully connected classifier. The first dense layer reduces the features to 256 dimensions, followed by Layer Normalization, ReLU

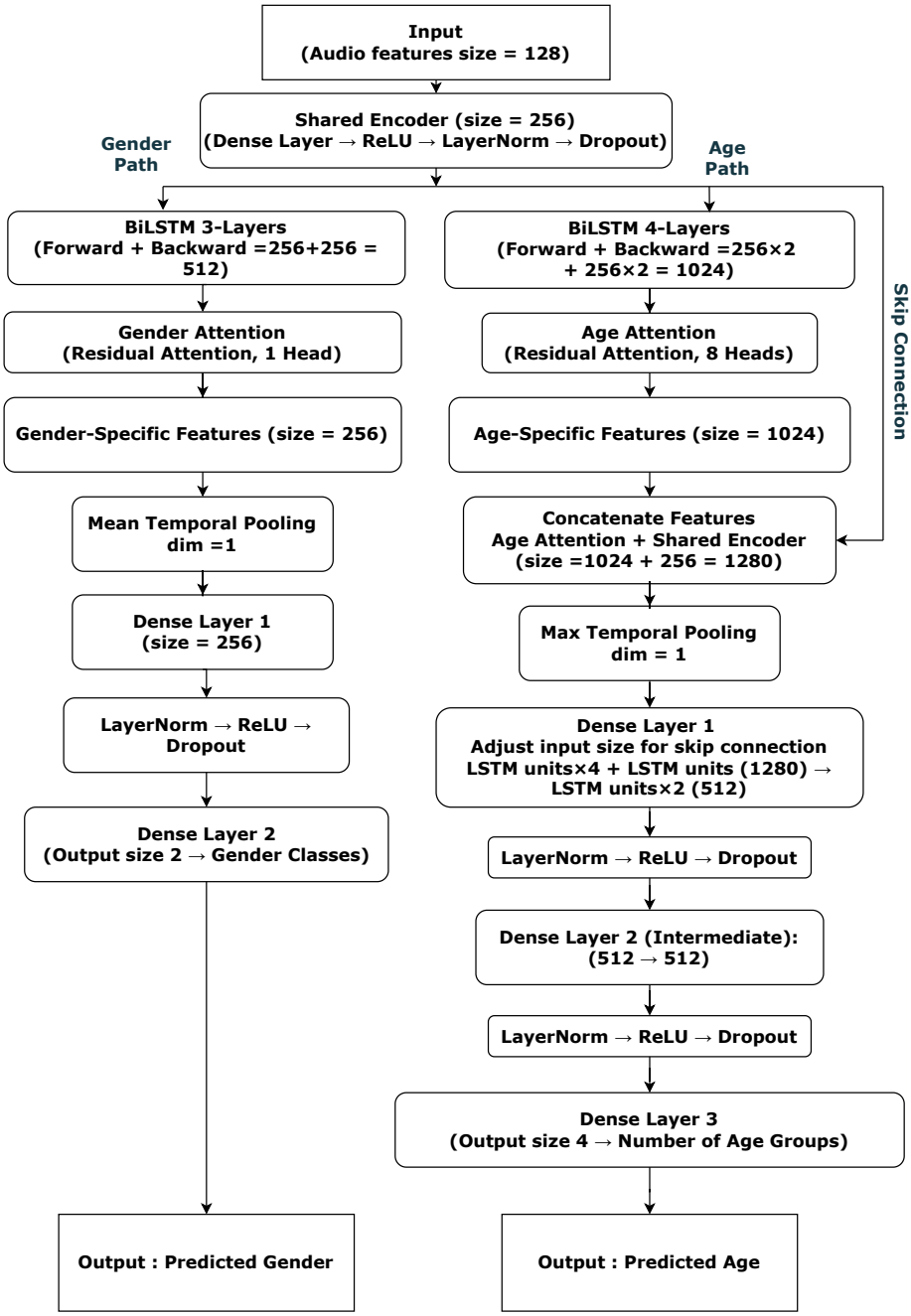


Figure 4.7: Workflow of proposed Modified BiLSTM model architecture

activation and dropout (0.4). The second dense layer produces the final logits, which are converted into probabilities over the two gender classes.

For supervision, a focal cross-entropy loss function is used. This reduces the effect of easy examples and gives more weight to harder or minority cases. For a training example i with true label y_i and predicted probability p_{i,y_i}, the loss is defined in Equation 4.2.

$$L_i = -\, w_g(y_i)\,(1 - p_{i,y_i})^{\gamma}\,\log(p_{i,y_i}) \tag{4.2}$$

Here, $w_g(y_i)$ denotes the class weight and γ represents the focusing parameter. The batch loss, averaged over all N samples, is given in Equation 4.3.

$$L_{\text{gender}} = \frac{1}{N}\sum_{i=1}^{N} L_i \tag{4.3}$$

Gradients of this loss with respect to model parameters ($\nabla_\theta L_{\text{gender}}$) are computed by backpropagation. The Adam optimizer with a learning rate of 1×10^{-4} is then used to update the parameters at each step. This process continues over multiple mini-batches and epochs until early stopping criteria are met.

4.5.1.3 Training of the Age Classification Model using Modified BiLSTM

The training of the age classification branch begins in the same way. Input features are passed through the shared encoder, producing a 256-dimensional representation normalized with Layer Normalization, activated with ReLU, and regularized with dropout (0.4).

The encoded features then enter the age branch, which uses a four-layer bidirectional LSTM with hidden size 512. This structure processes the sequence in both directions and dropout (0.4) is applied between layers. The LSTM output is then passed to a multi-head residual attention mechanism with eight heads. This mechanism allows the model to assign different importance to different time steps while keeping the original sequence information.

The output of attention is concatenated with the shared encoder features using a skip connection. This ensures that both the lower-level acoustic details and the higher-level contextual patterns are preserved. Max pooling is then applied across time, selecting the most dominant features.

The pooled representation is passed through a three-layer classifier. The first dense layer increases the feature size to 512, followed by Layer Normalization, ReLU and dropout (0.4). The second dense layer reduces the features to 256, followed by normalization, ReLU and dropout (0.2). The final dense layer gives the output for the age categories (teens, twenties, thirties, forties). A softmax function converts these logits into probabilities.

The age branch uses a customized loss function that combines standard cross-entropy, label smoothing, and a confidence penalty. For a training example i with

true label y_i and predicted probability distribution p_i, the loss is defined in Equation 4.4.

$$L_i = w_a(y_i)\, L_{\text{CE}}(i) + \alpha\, L_{\text{smooth}}(i) + \beta\, L_{\text{conf}}(i) \tag{4.4}$$

Here, $w_a(y_i)$ denotes the class weight for the true age group, L_{CE} represents cross-entropy loss, L_{smooth} redistributes some probability mass across other classes, and L_{conf} penalizes overly confident (sharp) predictions. In this work, $\alpha = 0.1$ and $\beta = 0.1$. The overall batch loss, averaged across all samples, is shown in Equation 4.5.

$$L_{\text{age}} = \frac{1}{N} \sum_{i=1}^{N} L_i \tag{4.5}$$

Backpropagation computes the gradients of this loss with respect to the age branch parameters ($\nabla_\theta L_{\text{age}}$). Parameter updates are performed using the Adam optimizer with a learning rate of 1×10^{-4}. Training continues over mini-batches until the validation loss stops improving.

4.5.2 Customized CNN for AI Detection

The AI detection model is designed to capture subtle differences between human and AI-generated voices by processing both spectral and temporal cues. Each audio clip, after preprocessing, is represented as a 128-dimensional log-Mel spectrogram spanning 5 seconds, resulting in approximately 157 time frames per clip at a 16 kHz sampling rate with standard windowing. This high temporal resolution allows the network to capture fine grained prosodic and spectral patterns.

4.5.2.1 Customized CNN Model Architecture:

The spectrogram first enters the convolutional feature extractor, which consists of three sequential 2D convolutional layers. Each convolutional layer is followed by a ReLU activation and a MaxPooling operation, reducing the spatial dimensions while emphasizing relevant spectral-temporal features. Dropout with a rate of 0.3 is applied after the convolutional blocks to prevent overfitting. This convolutional representation forms the foundational embedding for the AI detection task.

From the convolutional features, the output is flattened into a 38,912-dimensional vector. This flattened embedding is then forwarded into the classifier branch. The classifier consists of three fully connected layers: the first reduces dimensionality from 38,912 to 512 units, with ReLU activation and dropout (0.3); the second reduces dimensionality to 128 units, also with ReLU activation and dropout (0.3); and the final layer outputs a single logit representing the probability of the input being AI-generated, which is transformed through a Sigmoid activation to yield normalized probability scores. Overall, the architecture comprises approximately 20 million trainable parameters, efficiently balancing convolutional feature extraction and dense classification layers.

From a functional perspective, the model can process a batch of 32 five-second clips in a single forward pass, covering over 5,000,000 data points per batch and

producing class probabilities for each clip. The convolutional layers selectively emphasize discriminative spectral-temporal patterns, while the fully connected layers integrate these features across the entire clip to produce a global prediction. Feature visualization showed that the embeddings of human and AI-generated voices form well-separated manifolds, demonstrating strong representation learning.
The architecture maintains a temporal resolution of approximately 32 ms per frame after pooling, a high-dimensional spectral representation, and attention to discriminative patterns, making it robust to recording variations, pitch shifts, background noise, and synthetic speech artifacts. As illustrated in Figure 4.8, the voice enters the network as a spectrogram, is processed through hierarchical convolutional layers, flattened into a high-dimensional embedding, and classified through fully connected layers with a Sigmoid output. This carefully designed architecture underpins the reported final accuracy of 99.05% and balanced F1-score of 0.985, effectively combining spectral and temporal analysis to distinguish human voices from AI-generated speech.

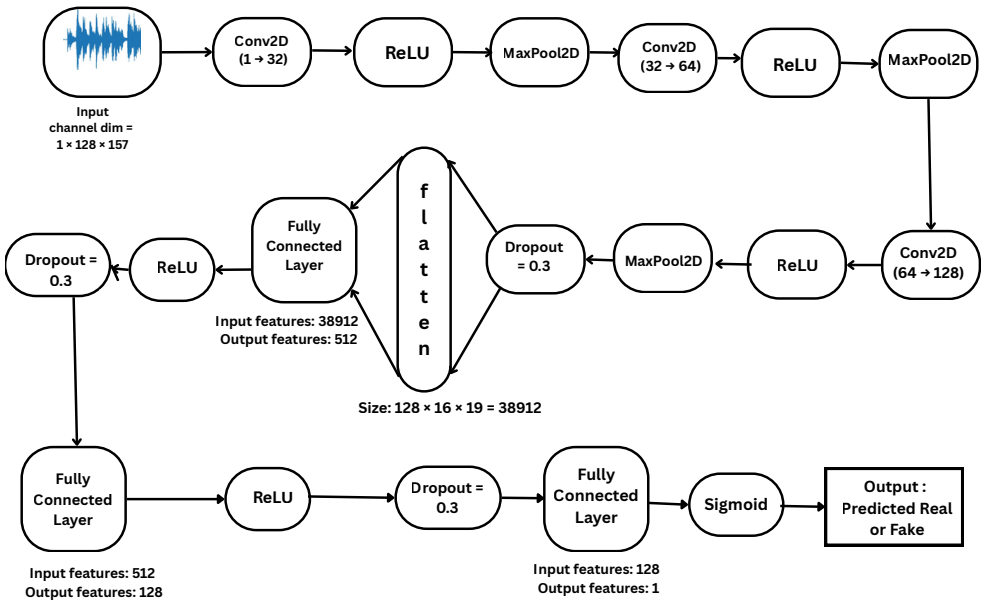


Figure 4.8: Workflow of proposed CNN Model Architecture

4.5.2.2 Training of the AI Detection Model

The training of the AI detection branch begins with mini-batches of speech samples, each represented as a 128-dimensional log-Mel spectrogram feature vector spanning 5 seconds. These input features are first processed by the CNN-based feature extractor, which applies three convolutional layers with ReLU activation and max-pooling, followed by dropout (0.3) to prevent overfitting. The resulting feature maps are flattened into a 38,912-dimensional vector, forming the stabilized embedding for classification.

This flattened representation is then directed into the AI detection-specific classifier branch. The features are sequentially processed through fully connected layers: first

from 38,912 to 512 units with ReLU activation and dropout (0.3), then from 512 to 128 units with ReLU activation and dropout (0.3), and finally to a single output unit with a Sigmoid activation producing a probability for the AI class. This architecture enables the model to learn complex spectral and temporal patterns distinguishing human voices from AI-generated voices, while maintaining efficient training with ~ 20 million trainable parameters.

Training is supervised using the Binary Cross-Entropy (BCE) loss function. For a training instance i with true label $y_i \in \{0, 1\}$ and predicted probability p_i, the BCE loss is defined in Equation 4.6.

$$L_i = -\left[y_i \log(p_i) + (1 - y_i) \log(1 - p_i)\right] \tag{4.6}$$

The batch-level loss, obtained by averaging over N samples, is shown in Equation 4.7.

$$L_{\text{batch}} = \frac{1}{N} \sum_{i=1}^{N} L_i \tag{4.7}$$

Backpropagation is then applied to compute the gradients $\nabla_\theta L_{\text{BCE}}$ with respect to the parameters θ of the network.

AI detection branch. The Adam optimizer updates the parameters adaptively using estimates of the first and second moments of the gradients, with a learning rate of 0.001, balancing stable convergence with efficient training.
The training process is conducted for 10 epochs with a batch size of 32, using 80% of unique voices for training and 20% for validation. Training dynamics indicate rapid convergence, with the model achieving training accuracy of 99.44% and validation accuracy of 99.05% at the final epoch. The small difference between training and validation loss (≈ 0.006) confirms good generalization and minimal overfitting.

Through repeated forward and backward passes across mini-batches, the model progressively learns to capture discriminative spectral and temporal features. The combination of convolutional feature extraction, sequential dense modeling and dropout-based regularization enables robust detection of AI-generated speech even under noisy conditions. The confusion matrix and classification report further demonstrate strong discrimination, with overall accuracy of 98.99% and F1-scores of 0.9852 for human and 0.9853 for AI, confirming reliable AI detection performance.

4.5.3 Regional Dialect Classification Model Architecture

The journey of each audio sample begins as a raw five-second waveform, representing the speech of a speaker from one of the ten districts. Before entering the model, the waveform is standardized—mean-normalized and amplitude-scaled—to ensure that variations in recording volume or microphone sensitivity do not affect subsequent processing.

4.5.3.1 Regional Dialect classification Model Architecture:

The proposed model is designed to classify Bengali voice samples by learning robust representations directly from raw audio waveforms. Each input sample is first loaded

and standardized to a fixed length, ensuring consistency across the dataset. This preprocessing step includes normalization to zero mean and unit variance, which stabilizes training and allows the network to focus on meaningful acoustic patterns rather than amplitude differences.

Once preprocessed, the audio waveform is fed into an initial convolutional block. This block applies a 1D convolution with a relatively large kernel size and stride, followed by batch normalization, ReLU activation and max pooling. This stage extracts low-level features such as edges in the waveform and short-term spectral patterns while reducing the temporal resolution to make subsequent processing more efficient.

The output of the initial convolution is then passed through a series of residual blocks. Each residual block consists of two convolutional layers, each followed by batch normalization and ReLU activation, along with a skip connection that allows the original input to bypass the convolutions. This design mitigates the vanishing gradient problem and enables the network to learn deeper hierarchical features without degradation. As the signal propagates through successive residual blocks, the number of channels increases, allowing the network to encode increasingly abstract and discriminative representations of the audio signal, capturing features relevant to dialect, speaker characteristics and subtle voice variations.

After the convolutional and residual processing, the network applies global average pooling, which compresses each feature map into a single representative value. This step reduces the feature dimensionality while preserving the most salient characteristics learned by the convolutional layers. The resulting vector is then passed through fully connected layers with ReLU activations and dropout regularization. Dropout is used to prevent overfitting by randomly deactivating neurons during training, ensuring that the model generalizes well to unseen samples.

Finally, the last fully connected layer maps the learned features to the output space, producing probabilities for each district class through the softmax function. The model is trained end-to-end using cross-entropy loss, optimizing the network to maximize the likelihood of correctly classifying each audio sample.

Overall, the architecture sequentially transforms raw audio into high-level representations by combining convolutional feature extraction, residual connections for deep hierarchical learning, global pooling and dense classification layers. This end-to-end design ensures that both local waveform patterns and broader temporal structures are captured effectively, resulting in a robust model for Bengali voice classification, as illustrated in Figure 4.9, which presents the workflow of the proposed Modified Residual Network model architecture.

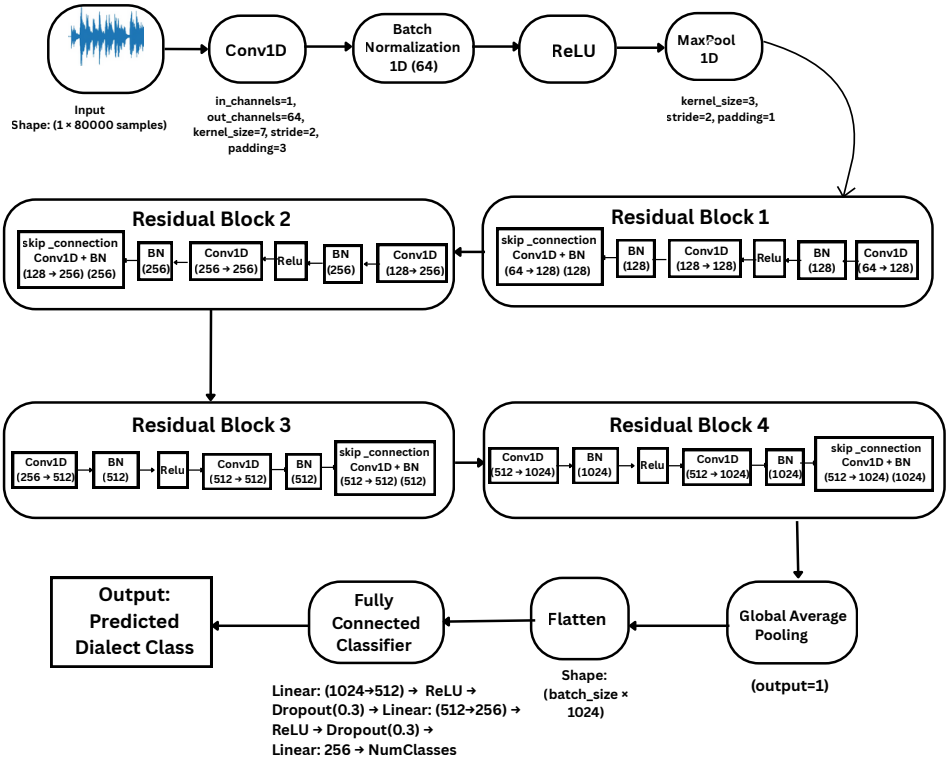


Figure 4.9: Workflow of proposed Modified Residual Network Model Architecture

4.5.3.2 Training of the Regional Dialect Model

The training of the regional dialect classification model begins with batches of raw audio waveforms, each standardized to a fixed length of 5 seconds (80,000 samples at 16 kHz). Each waveform is first normalized to zero mean and unit variance, ensuring numerical stability and consistent feature scaling across the dataset. For a waveform sample x_i, the normalization process is defined in Equation 4.8.

$$x_i' = \frac{x_i - \mu_{x_i}}{\sigma_{x_i} + \varepsilon} \tag{4.8}$$

Here, μ_{x_i} and σ_{x_i} denote the mean and standard deviation of the waveform, respectively, while ε is a small constant added to prevent division by zero.

The normalized waveform is then fed into the initial convolutional block of the Deep CNN. Here, a 1D convolution with 64 filters of kernel size 7 and stride 2 extracts local spectral patterns. Batch normalization is applied to stabilize the training, followed by a ReLU activation to introduce non-linearity. Max pooling with kernel size 3 and stride 2 reduces temporal dimensionality while retaining salient features.

Following the initial convolution, the model passes through a series of four residual blocks. Each residual block consists of two sequential 1D convolutional layers with kernel size 3, inter leaved with batch normalization and ReLU activation. To maintain information from earlier layers, skip connections are applied, either as identity

mappings or via 1×1 convolutions when the input and output dimensions differ. The residual operation for a block is defined in Equation 4.9.

$$y = \mathrm{ReLU}\big(F(x, W) + X_{\mathrm{shortcut}}\big) \tag{4.9}$$

Here, x is the input to the block, F represents the stacked convolutions and batch normalizations, and X_{shortcut} is the skip connection. These residual connections allow the gradient to flow directly through the network, mitigating vanishing gradient issues in deep architectures.

After the final residual block, global average pooling condenses the feature maps along the temporal dimension, producing a fixed-length 1024-dimensional vector that summarizes the most important dialectal characteristics. Dropout with a rate of 0.5 is applied to prevent overfitting, followed by a three-layer fully connected classifier. The first dense layer reduces the vector to 512 dimensions, followed by ReLU activation and dropout (0.3). The second dense layer further compresses the representation to 256 dimensions, and the final layer outputs corresponding to the number of dialect classes.

The model is supervised using the categorical cross-entropy loss. For a sample i with true one-hot label y_i and predicted probabilities $\hat{y}_i$, the loss is defined in Equation 4.10.

$$\mathcal{L}_i = -\sum_{c=1}^{C} y_{i,c} \log \hat{y}_{i,c} \tag{4.10}$$

The batch-level loss, obtained by averaging over N samples, is shown in Equation 4.11.

$$\mathcal{L}_{\mathrm{dialect}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_i \tag{4.11}$$

Gradients $\nabla_\theta \mathcal{L}_{\mathrm{dialect}}$ with respect to model parameters are computed via backpropagation. To prevent gradient explosion, gradient norms are clipped as described in Equation 4.12:

$$\|\nabla_\theta \mathcal{L}_{\mathrm{dialect}}\|_2 \leq 1 \tag{4.12}$$

The Adam optimizer updates the model parameters at each iteration, adjusting them according to the computed gradients and a learning rate of 1×10^{-3}. To further improve convergence, a `ReduceLROnPlateau` scheduler monitors the validation loss and reduces the learning rate when progress stagnates.

Training proceeds iteratively over multiple mini-batches and epochs, with periodic evaluation on the validation set to track the classification accuracy. Accuracy is computed as shown in Equation 4.13.

$$\text{Accuracy} = \frac{\#\text{ correctly predicted samples}}{\text{total samples}} \times 100 \tag{4.13}$$

The best-performing model is saved based on validation accuracy and lowest validation loss, ensuring optimal generalization to unseen dialectal speech.

4.5.4 Siamese Model Architecture

This section introduces the Siamese model architecture and provides a general idea of how it is designed. It gives an summary of the main parts of the model and the process used to learn meaningful embeddings that can be compared to measure similarity between inputs.

4.5.4.1 Siamese Model Architecture Design

The proposed Siamese model is designed to learn embedding representations of input audio features and to compare them in order to measure similarity between two samples. The architecture is divided into three main components: a shared encoder, an embedding projection network and a similarity scoring network.

The shared encoder processes the input audio feature sequence and extracts relevant representations. Depending on the configuration, the encoder may include a bidirectional LSTM with 1 to 4 layers, which captures sequential information in the input. An optional attention mechanism can also be applied. Two attention variants are supported: LightAttention, which applies a linear scoring function with softmax weighting and BiLSTMAttention, which introduces an additional BiLSTM followed by attention weighting. In both cases, weighted mean and standard deviation pooling are applied to form the attention feature vector. The final encoder representation is either the flattened sequence output alone or the concatenation of flattened features and attention features.

The embedding projection network then takes the features produced by the encoder and transforms them into a fixed length vector, called an embedding, that represents the input. This is achieved using a multilayer perceptron (MLP), which is a series of fully connected layers where each layer applies a linear transformation, followed by batch normalization, a ReLU activation and dropout to prevent overfitting. The last layer of the MLP outputs a vector of size emb (for example, 128 numbers), which is normalized using the L2 norm to ensure its length is 1. The resulting embedding is a compact and consistent representation of the input in a fixed dimensional space R^{emb} that can be used to compare different inputs.

The similarity scoring network then compares a pair of embeddings. Given two embeddings e_1 and e_2, a joint representation is constructed by concatenating e_1, e_2, their element wise absolute difference and their element-wise product, resulting in a vector in $R^{4 \cdot emb}$. This joint representation is passed through another MLP consisting of fully connected layers with batch normalization, ReLU activation and dropout and the final dense layer outputs a single value that is passed through the hyperbolic tangent function to produce a similarity score in the range $[-1, 1]$.

This overall architecture enables the model to project each input into a shared embedding space and to learn a similarity function that operates on the embeddings, making it suitable for verification or matching tasks, as illustrated in Figure 4.10.

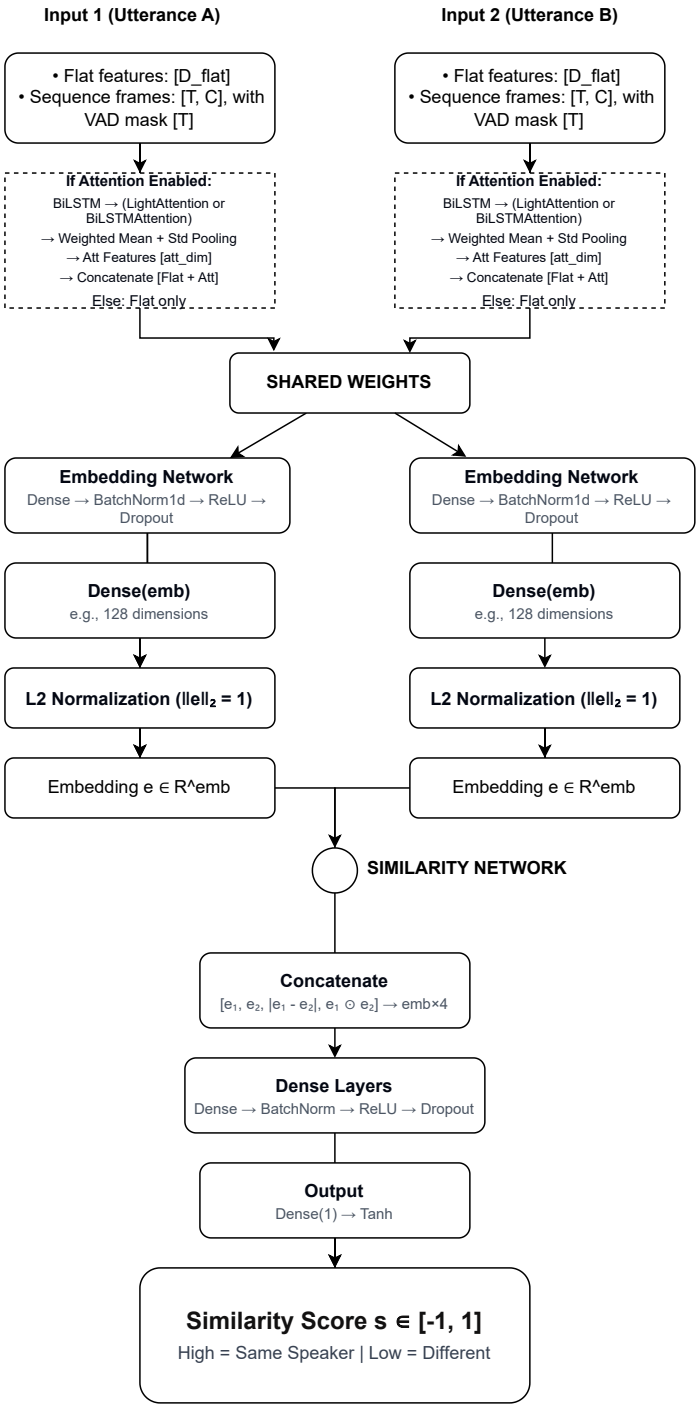


Figure 4.10: Workflow of proposed Siamese model architecture

4.5.4.2 Training of the Siamese Model

The training of the **Siamese model** implements a network that processes **pairs of utterances** to learn speaker similarity. Each pair consists of either two utterances from the same speaker (genuine pair) or two utterances from different speakers (impostor pair). This setup allows the model to directly optimize a similarity metric rather than performing traditional classification.

Each utterance is represented by precomputed **flat feature vectors** and optionally by **dynamic frame sequences** for the attention branch. Dynamic sequences are normalized using training set statistics. For a given channel x_i, normalization is defined in Equation 4.14.

$$x_i' = \tanh\left(\frac{x_i - \mu_i}{\sigma_i + \epsilon}\right) \tag{4.14}$$

Here, μ_i and σ_i are the mean and standard deviation of the channel across the training set, and ϵ is a small constant to prevent division by zero. Selected channels such as pitch (F0) and formants undergo a logarithmic transformation $\log(1 + x_i)$ to stabilize their distribution.

In the Siamese model, both utterances pass through the **same embedding network**, consisting of fully connected layers with batch normalization, ReLU activation and dropout. If the attention branch is enabled, dynamic sequences are pooled using either light attention or BiLSTM-based attention to produce fixed size embeddings that capture temporal patterns. Embeddings from the flat and attention branches are concatenated and L2 normalized to form a 128 dimensional vector for each utterance.

The **similarity head** combines embeddings via concatenation, element-wise absolute difference, and Hadamard product, outputting a similarity score $s \in [-1, 1]$. The model is supervised using a **focal loss adapted for similarity scores**, where the score is first mapped to a probability as shown in Equation 4.15.

$$p_i = \frac{s_i + 1}{2} \tag{4.15}$$

The batch loss is computed as a weighted binary cross-entropy with a focusing factor, defined in Equation 4.16.

$$L = \frac{1}{N} \sum_{i=1}^{N} \alpha(1 - p_i)^{\gamma} \, \mathrm{BCE}(p_i, y_i) \tag{4.16}$$

Here, $y_i \in \{0, 1\}$ indicates whether the pair is genuine or impostor, and α and γ are hyperparameters controlling the focus on hard examples.

Gradients $\nabla_\theta L$ are computed by backpropagation, with **gradient clipping** applied to a maximum norm of 1.0 to prevent explosion. Model parameters are updated using the AdamW optimizer. The learning rate is controlled by a **Cosine Annealing Warm Restarts (CAWR) scheduler**, defined in Equation 4.17.

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left(1 + \cos \frac{T_{\text{cur}}}{T_i} \pi\right) \tag{4.17}$$

Here, η_t is the learning rate at the current step, $\eta_{\min}$ is the minimum learning rate, $\eta_{\max}$ is the initial learning rate, T_i is the length of the current cycle in steps, and T_{cur} is the number of steps elapsed since the start of the current cycle. This scheduler allows the learning rate to periodically restart, improving convergence over long training.

Training proceeds iteratively over mini batches and epochs, with evaluation on a **fixed validation set of genuine and impostor pairs**. The best performing model is saved based on the highest validation accuracy, ensuring optimal generalization to unseen speakers.

Chapter 5

Model Evaluation

5.1 Performance Evaluation of Age and Gender Classification Model

This section is a review of different evaluation methods that examines how well the proposed models identify gender and age in voice samples in the bengali voice dataset. The assessment methods target examining the classification process effectiveness through its accuracy rates and precision together with reliability checks. The metrics demonstrate how well the model detects gender classes and age divisions yet reduces errors with consistent performance throughout the dataset.

5.1.1 Evaluation Metrics for gender and age-based classification from voice samples

5.1.1.1 Precision

Precision measures the relationship between correct gender/age predictions (true positives) and the total number of predicted samples (true positives + false positives) in each category. The model's reliability in generating appropriate category classifications is evaluated using this metric, as defined in Equation 5.1.

$$\text{Precision}_{\text{gender/age}} = \frac{\text{Correctly Predicted Samples for a gender or age}}{\text{Total Samples Predicted for that gender or age}} \tag{5.1}$$

5.1.1.2 Accuracy

To classify gender and age, accuracy measures the proportion of correctly classified voice samples relative to the total number of samples in the dataset. The model's combined performance for gender and age identification is evaluated using this metric, as defined in Equation 5.2.

$$\text{Accuracy} = \frac{\text{Number of correctly classified gender and age samples}}{\text{Total number of samples}} \tag{5.2}$$

5.1.1.3 Recall

The Recall metric evaluates a model's ability to correctly identify true positives from a specific category, relative to all actual samples in that category (true positives + false negatives), effectively measuring how well false negatives are avoided. The model's prediction accuracy relies heavily on proper representation of all age and gender groups, as defined in Equation 5.3.

$$\text{Recall}_{\text{gender/age}} = \frac{\text{Correctly Predicted Samples for a gender or age}}{\text{Total True Samples for that gender or age}} \tag{5.3}$$

5.1.1.4 F1 Score

While evaluating a model's performance, the F1 Score provides a balanced evaluation by computing the harmonic mean of precision and recall, making it effective for handling class imbalances. The F1 Score is particularly useful for multiclass classification problems such as age group prediction, as defined in Equation 5.4.

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.4}$$

5.1.1.5 Loss Functions:

For this age and gender classification model, a particular loss functions during model optimization for tasks which involved gender and age classification. These loss functions is designed to handle class imbalances and improve the model's ability to learn. The two loss functions that is used here are- **balanced focal loss function** and **age-focused loss function**. In both the loss function cross entropy loss is used indirectly as base loss function in an extended way.

- **Balanced focal loss function** Both the gender and age classification tasks use the Balanced Focal Loss function as their loss function. This approach helps the model allocate greater attention to difficult samples while reducing the impact of easily classified examples, particularly in minority categories. The Balanced Focal Loss is an extended weighted version of the standard focal loss, defined in Equation 5.5:

  $$\text{Balanced Focal Loss} = -\alpha_w (1 - p_t)^\gamma \log(p_t) \tag{5.5}$$

  Here, p_t represents the predicted probability of the correct class, and the logarithmic term corresponds to the standard cross-entropy loss, $-\log(p_t)$. The weighting factor α_w is a class-specific scaling term computed as the product of the focal loss weight α_t and the class weight, assigning higher values to underrepresented classes to balance the contribution of each category during training.

- **Age Focused loss function** Customized loss function is used for age classification which is basically combined with cross entropy loss function with additional penalties like- Label smoothing and confidence penalty.This function creates a balance between age group learning while reducing dataset class

dominance overfitting. The Age-Focused Loss for sample i is defined in Equation (5.6), which combines weighted cross-entropy, label smoothing, and a confidence penalty to improve age classification performance.

$$L_{age}(i) = w_a(y_i)L_{CE}(i) + \alpha L_{smooth}(i) + \beta L_{conf}(i) \tag{5.6}$$

Here, $w_a(y_i)$ denotes the class-specific age weight, $L_{CE}(i)$ is the cross-entropy loss, $L_{smooth}(i)$ is the label smoothing component, and $L_{conf}(i)$ represents the confidence penalty term. The hyperparameters α and β control the influence of the smoothing and confidence penalty components, respectively.

Besides, there is another loss, i.e., the total training loss, which is the combination of both gender and age losses. The overall training objective combines the contributions from both gender and age classification tasks, as defined in Equation (5.7). Here, L_{total} represents the total loss for a single training sample, L_{gender} is the loss associated with gender classification, and L_{age} is the Age-Focused Loss defined in Equation (5.6). By summing these two components, the model jointly optimizes for accurate prediction of both attributes, allowing the network to balance learning across gender and age tasks.

$$L_{total} = L_{gender} + L_{age} \tag{5.7}$$

5.1.1.6 Macro Average and Weighted Average

The **macro average** calculates the arithmetic mean of precision, recall, and F1 Score across all classes, treating each class equally regardless of its frequency. This evaluation metric provides an overall measure of model performance across all classes. The macro-averaged precision, recall, and F1 Score can be expressed as:

$$\mathrm{F1}_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C}\mathrm{F1}_c \tag{5.8}$$

where C is the total number of classes and and $\mathrm{F1}_c$ are the metrics for class c.
The **weighted average**, on the other hand, takes the number of instances in each class into account while computing precision, recall, and F1 Score. It gives higher weight to more frequent classes and can be influenced by class imbalance. Weighted averages are computed as:

$$\mathrm{F1}_{\text{weighted}} = \frac{\sum_{c=1}^{C} n_c \,\mathrm{F1}_c}{\sum_{c=1}^{C} n_c} \tag{5.9}$$

where n_c denotes the number of samples in class c. While comparing models, the macro average is often preferred as it treats all classes equally, providing a more balanced evaluation across imbalanced datasets.

The authentication systems within **Bengali Voice Signature Authentication for Financial System** demands finding optimal precision-recall relationships that promote safety measures and user-friendly systems at once. The highest priority lies with precision because it reduces the potential for false positive outcomes that stop unauthorized users from being mistakenly authenticated thus protecting against

financial fraud. While both precision and recall are necessary factors to prevent security issues. High recall guards against false negatives by stopping valid users from being blocked, thus keeping user experience positive and serviceable.Since both evaluation metrics remain crucial, the F1 Score provides a balanced assessment method that maximizes authentication accuracy.

Now, in the context of **gender and age classification**, which is the base or fundamental step for identity verification,by ensuring precise detection of speaker attributes while maintaining authentication reliability which is strengthened by accurate attribute results.For continuous authentication success the system needs a strong capability to detect various expressions of gender throughout all vocal variations. Appropriate detection of user age represents a critical factor in fraud detection since age-related discrepancies between voice signatures and claimed identities function as defensive tools against fraud.A financial authentication system needs to be both precise to block unapproved access attempts and helpful enough to authenticate real users while keeping its fraud prevention and user convenience as the main goals.

5.1.2 Statical Analysis of Design Solutions for age and gender model

In this section, the performance evaluation of the proposed model is discussed. The evaluation of the design solutions relies on precision along with recall, F1-score, confusion matrices and training loss/accuracy graph metrics for performance assessment.

5.1.2.1 Hyperparameter tuning for Modified BiLSTM model for Gender and Age Classification

Table 5.1 summarizes the hyperparameter tuning experiments for gender classification. The model configurations varied in terms of LSTM units, dropout rate, learning rate, batch size, and layer depth to evaluate their impact on classification accuracy. The current configuration, with 256 LSTM units, a dropout rate of 0.4, and a learning rate of 0.0001, achieved the highest accuracy of 98%, outperforming all other experimental setups. This indicates that moderate regularization and optimal learning rate balancing enhance model generalization in gender recognition tasks. As shown in Table 5.1, the optimal configuration achieved an accuracy of 98%, demonstrating that tuning the dropout and learning rate significantly influences gender classification performance.

Table 5.2 presents the outcomes of the hyperparameter tuning experiments for age classification. Various combinations of LSTM units, dropout rates, learning rates, and layer depths were tested to identify the most effective model configuration. The best performance, with an accuracy of 95%, was achieved using 512 LSTM units, a dropout rate of 0.4, and a learning rate of 0.0002. This suggests that moderate regularization and sufficient network depth enhance the model's ability to generalize across diverse age groups, while overly complex architectures lead to reduced accuracy due to overfitting. As summarized in Table 5.2, the optimal configuration

Experiment	Units	Dropout	LR	Batch Size	Layers	Accuracy
Current	256	0.4	0.0001	64	3	**98%**
Exp-G1	128	0.3	0.0005	32	2	93%
Exp-G2	512	0.4	0.0001	64	3	94%
Exp-G3	256	0.2	0.0001	128	4	93%
Exp-G4	384	0.3	0.00005	64	3	86%
Exp-G5	256	0.5	0.0002	32	3	84%

Table 5.1: Gender Classification Experiments

attained 95% accuracy, indicating that careful balancing of dropout and learning rate parameters substantially improves age classification performance.

Experiment	Units	Dropout	LR	Batch Size	Layers	Accuracy
Current	512	0.4	0.0002	64	4	**95%**
Exp-A1	256	0.3	0.0003	32	3	88%
Exp-A2	768	0.4	0.0001	64	5	93%
Exp-A3	512	0.2	0.0002	128	4	89%
Exp-A4	640	0.35	0.00015	64	4	91%
Exp-A5	512	0.5	0.0004	48	6	86%

Table 5.2: Age Classification Experiments

5.1.2.2 Gender Classification using Modified BiLSTM model

The performance of the **Modified BiLSTM** model for gender classification from the Table 5.3 it is seen that it has achieved a high accuracy of 98% and consistent high precision and recall for both genders. For male its precision and recall are 0.97 and 1.00 respectively and for females its precision and recall are 1.00 and 9.8 respectively. It clearly shows how good the model is for handling gender classification. Now in the confusion matrix of Figure 5.1 it can be seen that only 32 males misclassified as females and 280 females were wrongly classified as males. So, this model's capability to differentiate between the genders is very high. In addition, the loss-accuracy curves in Figure 5.3 shows that the gender classification loss decreases rapidly and stabilizes within the first few epochs. This consistently low loss demonstrates that the model effectively learns gender-related features without signs of overfitting. Similarly, the accuracy rises sharply, reaching around 97–98% within the first 10 epochs, after which it remains stable. These results indicate that the model is highly robust for gender-based identification.

5.1.2.3 Age-Based Classification using Modified BiLSTM model

Modified BiLSTM model for age based classification,if one look into the performance metrics of this model from Table 5.4 it is observed that its accuracy is 95% and the F1 Scores for teens, twenties, thirties and forties are 0.93, 0.90, 0.98 and 1.00 respectively. And the precision and recall ranges from 0.95-1.00 and 0.91 to 1.00 respectively. This metrics clearly shows how good the model performs for age classification and specially for forties age group. From the Figure 5.2 it can be observed

that the confusion matrix that there are misclassifications between closely related classes like twenties and teens but very minimal misclassifications in the forties age group with all samples classified correctly. In Figure 5.4 it can be observed that the loss values are higher compared to gender classification, which reflects the greater complexity of the multi-class age classification task. On the other hand, the training accuracy gradually improved, reaching around 95% after nearly 25 epochs.

Class	Precision	Recall	F1-score	Support
male_masculine	0.97	1.00	0.98	8711
female_feminine	1.00	0.98	0.99	11289
accuracy	-	-	0.98	20000
macro avg	0.98	0.99	0.98	20000
weighted avg	0.98	0.98	0.98	20000

Table 5.3: Gender Classification Report (Modified BiLSTM model)

Class	Precision	Recall	F1-score	Support
teens	0.95	0.91	0.93	5000
twenties	0.91	0.90	0.90	5000
thirties	0.97	0.98	0.98	5000
forties	1.00	1.00	1.00	5000
accuracy	-	-	0.95	20000
macro avg	0.95	0.95	0.95	20000
weighted avg	0.95	0.95	0.95	20000

Table 5.4: Age Classification Report (Modified BiLSTM model)

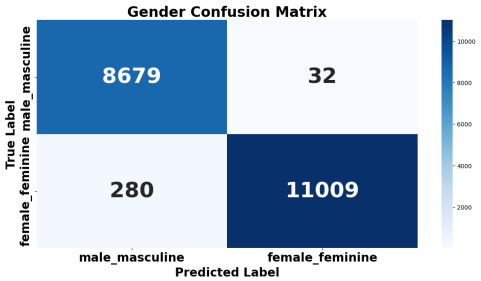


Figure 5.1: Gender Classification Confusion Matrix (Modified BiLSTM)

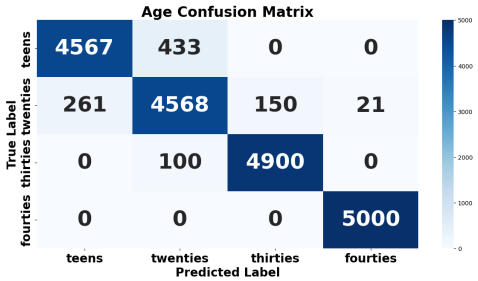
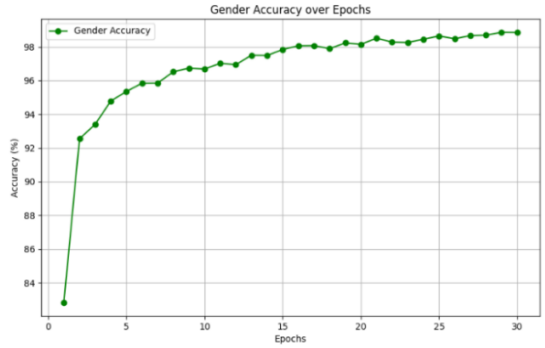


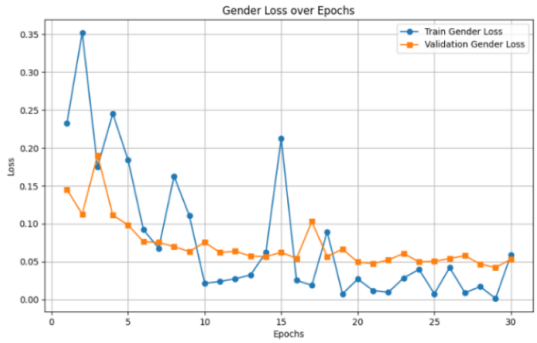
Figure 5.2: Age Classification Confusion Matrix (Modified BiLSTM)

5.1.3 Final Design Adjustment

In this section, the main changes made to our model design is examined, which encompass the incorporation of early stopping, adjusting the number of layers and investigation of the Leaky ReLU ultimately using ReLU activation function.

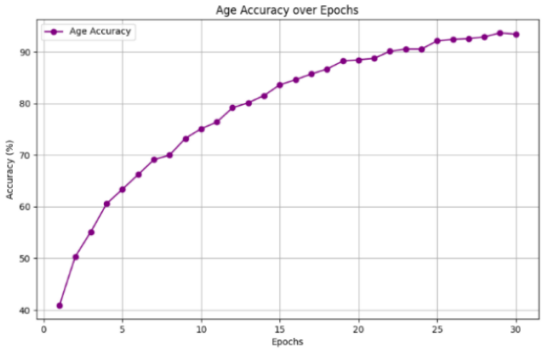


(a) Accuracy over epochs

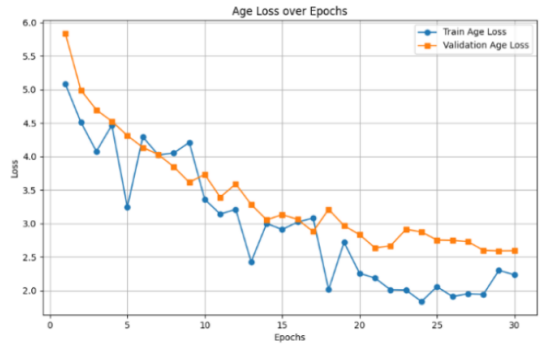


(b) Training and validation loss over epochs

Figure 5.3: Loss and accuracy graphs for gender classification using Modified BiLSTM model



(a) Accuracy over epochs



(b) Training and validation loss over epochs

Figure 5.4: Loss and accuracy graphs for age classification using Modified BiLSTM model

5.1.3.1 Early Stopping

Early stooping is employed to prevent overfitting throughout the training process. The purpose of using early stopping is that it can terminate the training process when the validation loss stops improving after a specified number of epochs that helps avoid overfitting; ensuring that the model is not trained beyond its optimal level. In the age-gender classification model, a very satisfactory result is received after 29 epochs while using early stopping. The absence of early stopping would have enabled the model to lead to a poor generalization since without it, there was a high risk of overfitting by learning noise present in the training data.

5.1.3.2 Effect of Layer Adjustments

It is known that the more layers in a model, the more complex patterns but the more good performance. During the model designingit is found in observation that the number of layers generally improves performance. However, for a smaller dataset, increasing the number of layers may lead to the risk of overfitting. On the contrary, decreasing layers can lead to underfitting that can result in failure in capturing integral patterns in data. Satisfactory results is found by applying three layers for gender classification and four layers for age classification in the model that balanced the performance and training affirming that the model was neither underfitting nor

overfitting.

5.1.3.3 ReLU vs. Leaky ReLU Activation

It is initially investigated that the activation function Leaky ReLU as a substitute for the standard ReLU activation function. While exploration, Leaky ReLU solves the "dying RelLU" issue where neurons may become inactive throughout the training process. However after analyzing the performance of both ReLU and Leaky ReLU, a good performance of ReLU is found for which this activation function is used in the model. Additionally, ReLU improved the computational functionality and assists the model to learn effectively even without the inactive neurons. Consequently, ReLU enhances the stability and increased training efficiency.

5.1.3.4 Dropout Regularization

To avoid overfitting, dropout layers were added throughout the architecture. For both gender and age classification, a dropout rate of 0.4 is applied after LSTM layers and fully connected layers. In the age classifier, the final dense layers uses a slightly reduced dropout rate of 0.2, which helps preserve finer details important for multi-class prediction. This careful tuning of dropout ensures that the models generalizes well without losing important information.

5.1.3.5 Hidden Size Tuning

The hidden size of the LSTM layers are adjusted to match the task complexity. In the gender branch, a hidden size of 256 is seen sufficient, since the task was binary classification. In contrast, the age branch requires a larger hidden size of 512 to capture the richer patterns needed for multi-class prediction. This adjustment improved the capacity of the model without causing instability.

5.1.3.6 Optimizer and Learning Rate

Both models uses the Adam optimizer with a learning rate of 1×10^{-4}. This choice balances stable convergence with training efficiency. A higher learning rate risks instability, while a lower rate slows training. With this configuration, both gender and age models reaches high accuracy within a reasonable number of epochs.

5.1.4 Comparisons and Relationships for Age and Gender Classification Model

In this study, three other models are evaluated alongside modified BiLSTM for Bengali voice recognition to classify gender and age. These models are RNN, LSTM and CNN BiLSTM which are trained and tested on the same dataset under the same conditions applied for modified BiLSTM. In the analysis, accuracy, precision, recall, and F1-score are used in the evaluation metrics to compare the results with the proposed modified BiLSTM.

5.1.4.1 Gender Classification Performance Across Models

The performance of different models for gender classification are evaluated using precision, recall, F1-score and accuracy metrics. Tables 5.5, 5.6, and 5.7 summarize the classification reports for the RNN, LSTM, and CNN BiLSTM models, respectively. A comparative overview of these models is presented in Table 5.8, highlighting the superior performance of the LSTM and proposed modified BiLSTM model in capturing gender-specific voice patterns. Overall, the results indicate notable differences in model effectiveness, with sequence-based architectures achieving higher accuracy than hybrid CNN-RNN approaches.

Class	Precision	Recall	F1-score	Support
male_masculine	0.61	1.00	0.76	9102
female_feminine	1.00	0.47	0.64	10898
accuracy	-	-	0.71	20000
macro avg	0.80	0.73	0.70	20000
weighted avg	0.82	0.71	0.69	20000

Table 5.5: Gender Classification Report (RNN model)

Class	Precision	Recall	F1-score	Support
male_masculine	0.95	0.98	0.97	8588
female_feminine	0.98	0.96	0.97	11412
accuracy	-	-	0.96	20000
macro avg	0.96	0.96	0.96	20000
weighted avg	0.96	0.96	0.96	20000

Table 5.6: Gender Classification Report (LSTM model)

Class	Precision	Recall	F1-score	Support
male_masculine	0.52	1.00	0.68	9020
female_feminine	0.99	0.22	0.36	10880
accuracy	-	-	0.57	20000
macro avg	0.75	0.61	0.52	20000
weighted avg	0.77	0.57	0.50	20000

Table 5.7: Gender Classification Report (CNN BiLSTM model)

Accuracy Comparison: The results as shown in Table 5.8, illustrate the gender classification report for the four models all together where LSTM and Modified BiLSTM models perform superior to other architectures since these models achieves over 95% accuracy. This higher accuracy is the result of the exponential capability of the models in capturing the temporal dependencies present in speech data. Contrarily, RNN and CNN-BiLSTM model performs significantly lower accuracy about 71% and 57% that indicates the challenges of identifying the gender specific patterns from voice features.

Models	Precision	Recall	F1-Score	Accuracy
Our BiLSTM	0.98	0.99	0.98	0.98
CNN BiLSTM	0.77	0.57	0.50	0.57
RNN	0.82	0.71	0.69	0.71
LSTM	0.98	0.98	0.98	0.98

Table 5.8: Gender-Based Model Comparison

Recall and Precision Analysis: In the context of the recall, Modified BiLSTM and LSTM both achieved strong ratings about 99% and 96% respectively. However, CNN-BiLSTM performed high precision of 77% but it struggled with lower recall indicating the failure of detecting female class in a significant portion. This discrepancy points to an imbalance in which the model missed many relevant cases despite having higher prediction confidence. In the case of the RNN model, it showed a recall of 71% and precision of 82% suggesting a greater rate of false positives compared to the leading models.

F1-Score Insights: The overall gender-based model performance is visualized in both a line plot and a bar chart, as shown in Figure 5.5 and Figure 5.6, respectively. The line plot highlights the trends in precision, recall and F1-score across models, while the bar chart provides a clear comparison of model performance metrics for easier interpretation.

The overall gender-based classification model performance is visualized in both a line plot and a bar chart, as shown in Figure 5.5 and Figure 5.6, respectively. The line plot highlights the trends in precision, recall and F1-score across models, while the bar chart provides a clear comparison of model performance metrics for easier interpretation.

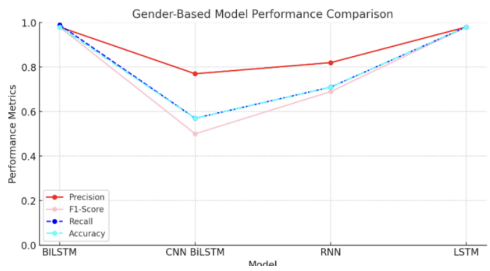


Figure 5.5: Gender-based model performance comparison.

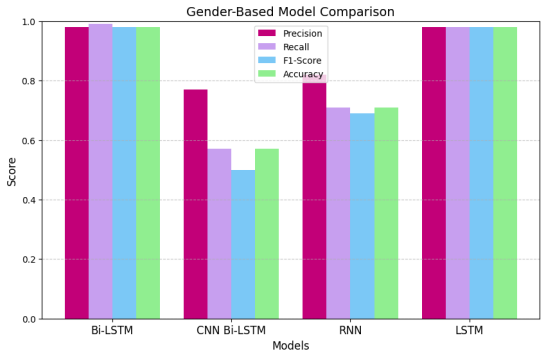


Figure 5.6: Gender-based model performance comparison barchart.

Key Observations:

- The models Modified BiLSTM and LSTM performed top where they achieved the highest position of accuracy, recall, precision and F1-score, rendering them particularly appropriate for gender classification in Bengali voice recognition systems.

- Despite having high precision of CNN-BiLSTM model, it struggled from low recall, reducing its reliability of feature extraction.
- RNN revealed diminished precision and recall, leading in less precise predictions and classification.

Recommended Model: From the overall analysis, the proposed Modified BiLSTM model is chosen as the most effective model since it performs better than the other leading models across all evaluation metrics. Although LSTM follows closely, because of its unidirectional construction, it is slightly less effective. However, RNN and CNN-BiLSTM are not recommended in this gender classification since they displayed lower accuracy and F1-score indicating inconsistent performance.

5.1.4.2 Age Classification Performance Across Models

Similarly as gender classification, the performance of various models for age classification are also evaluated using precision, recall, F1-score and accuracy metrics. Tables 5.9, 5.10, and 5.11 present the detailed classification reports for the RNN, LSTM and CNN BiLSTM models, respectively. A comparative overview of these models is summarized in Table 5.12, highlighting that the LSTM and BiLSTM models achieve the highest accuracy, whereas the CNN BiLSTM model shows limited performance. Overall, these results illustrate significant differences in model effectiveness for capturing age-specific voice patterns.

Class	Precision	Recall	F1-score	Support
teens	0.34	0.52	0.41	5000
twenties	0.00	0.00	0.00	5000
thirties	0.22	0.50	0.31	5000
forties	0.45	0.09	0.15	5000
accuracy	-	-	0.28	20000
macro avg	0.25	0.28	0.22	20000
weighted avg	0.25	0.28	0.22	20000

Table 5.9: Age Classification Report (RNN model)

Class	Precision	Recall	F1-score	Support
teens	0.79	0.97	0.87	5000
twenties	0.94	0.67	0.78	5000
thirties	0.93	0.94	0.93	5000
forties	0.96	1.00	0.98	5000
accuracy	-	-	0.90	20000
macro avg	0.90	0.89	0.89	20000
weighted avg	0.90	0.89	0.89	20000

Table 5.10: Age Classification Report (LSTM model)

Accuracy Comparison As shown in the table 5.12, the best performing model for age classification is the proposed Modified BiLSTM model with an accuracy of 95%. However, the LSTM model having 89% accuracy maintained good performance but

Class	Precision	Recall	F1-score	Support
teens	0.14	0.06	0.08	5000
twenties	0.33	0.70	0.44	5000
thirties	0.16	0.23	0.19	5000
forties	0.00	0.00	0.00	5000
accuracy	-	-	0.25	20000
macro avg	0.16	0.25	0.18	20000
weighted avg	0.16	0.25	0.18	20000

Table 5.11: Age Classification Report (CNN BiLSTM model)

Models	Precision	Recall	F1-Score	Accuracy
Our BiLSTM	0.95	0.95	0.95	0.95
CNN BiLSTM	0.16	0.25	0.18	0.25
RNN	0.25	0.28	0.22	0.28
LSTM	0.90	0.89	0.89	0.89

Table 5.12: Age-Based Model Comparison

slightly underperformed compared to Modified BiLSTM. On the contrary, RNN and CNN-BiLSTM performs bad in accuracy of 28% and 25% respectively, making them challenged in extracting age grouped speech features.

Recall and Precision Analysis In terms of recall, Modified BiLSTM achieved the highest recall of 95% and LSTM of 89% indicating that these models significantly classify age groups with a very minor misclassification where it is clear from the table 5.12 that RNN and Modified BiLSTM faces difficulties in recall scoring 28% and 25% demonstrating the higher misclassification rate. Also they have the lowest precision (16%) highlighting frequent errors in recognizing age groups.

F1-Score Evaluation From the evaluation metrics table 5.12, Modified BiLSTM achieved the highest balance between recall and precision since it has the highest F1-score of 95%. LSTM also displayed well with an F1-score of 89%. But RNN and CNN-BiLSTM models showed the lowest F1-scores of 22% and 18% respectively, demonstrating their lacking in accurate classification.

The age-based model performance is illustrated through both a line plot and a bar chart, as shown in Figure 5.7 and Figure 5.8, respectively. The line plot depicts trends in precision, recall, and F1-score across different models, while the bar chart provides a clear visual comparison of performance metrics for each model.

The combined gender–age model performance is illustrated using both a line plot and a bar chart, as shown in Figure 5.9 and Figure 5.10, respectively. The line plot highlights trends in precision, recall, and F1-score across all gender–age categories, while the bar chart provides a clear visual comparison of performance metrics for easier interpretation.

Key Observations

- Modified BiLSTM performs best with the best results in accuracy, recall, pre-

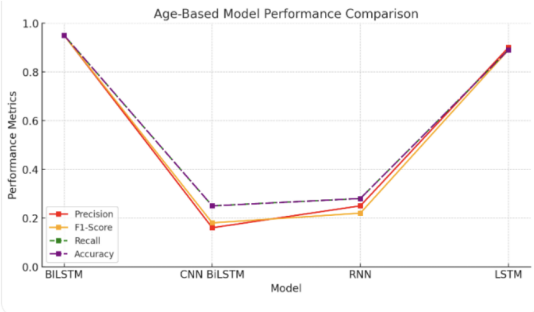


Figure 5.7: Age-based model performance comparison.

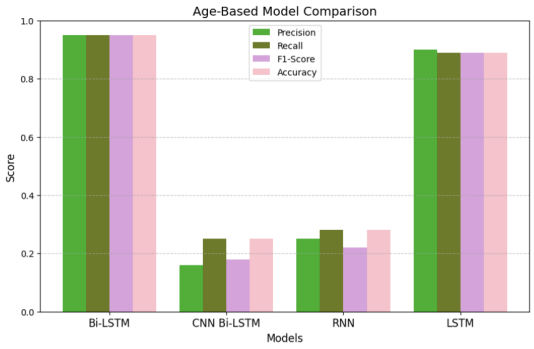


Figure 5.8: Age-based model performance comparison bar-chart.

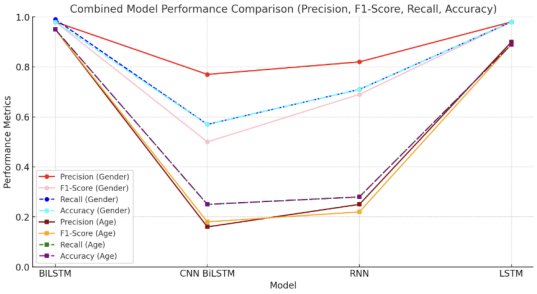


Figure 5.9: Gender-Age based model performance comparison.

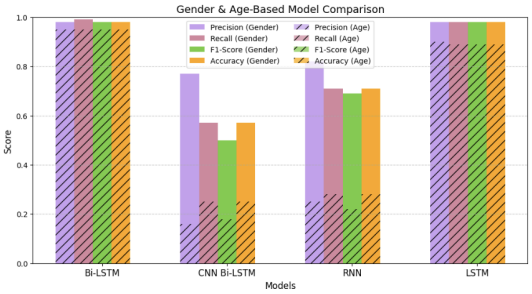


Figure 5.10: Gender-Age based model performance comparison barcharts.

cision and F1-score.

- Though LSTM displays strong performance, it is slightly under performing compare to the proposed Modified BiLSTM for its unidirectional processing.
- RNN and CNN-BiLSTM model performs very weak.

Recommended Model The top performing and most effective model for age classification is the proposed Modified BiLSTM since it outperforms all other models across all key metrics. Due to the unidirectional approach, although LSTM has also strong performance, it is minorly less effective than Modified BiLSTM model. However, RNN and CNN BiLSTM models are not recommended since they have unreliable classification performance due to their weak result in recall.

5.1.5 Validation of Modified BiLSTM for Gender and Age Classification Models

The training of the gender and age classification models is monitored for 30 epochs to check for overfitting or underfitting. For the gender model, both training loss and validation loss decrease steadily and the validation loss follows the training loss closely as seen in Figure 5.3. Accuracy improves quickly to above 95% within the first few epochs and later reached 98–99%. Since there are no clear gap between training and validation results, the gender model shows good generalization to unseen data.

The age model shows slow convergence because it has to handle multiple classes. From the graph Figure 5.4 it can be observed that training loss decreases from 5.08 in the first epoch to below 2.0 by epoch 25, while validation loss dropped from 5.83 to about 2.6. Validation accuracy increased gradually and passed 90% by the end of training. At some points, such as epoch 18, training loss is lower than validation loss, which suggested mild overfitting. However, validation accuracy continued to increase, which shows that overfitting is controlled and causes not harm to generalization. Regularization and spectrogram-based augmentation helped maintain this balance in this case.

Both models uses the Adam optimizer with a learning rate of 1×10^{-4} and batch size 64. Training is iniitially planned for 100 epochs but early stopping stopped the process after 30 epochs when validation loss stopped improving. This prevented overfitting and reduced extra computation. Regularization techniques are also applied at different levels. Layer Normalization kept feature distributions stable, dropout (0.4 and 0.2 depending on layer depth) reduced dependency between neurons and spectrogram-based augmentation (time masking, frequency masking noise injection and age-specific pitch/time changes) reduces the gap between training and validation performance.

The validation results confirms that the chosen optimization and regularization methods works well. In the gender model, dropout and normalization kept validation accuracy close to training accuracy. In the age model, extra mechanisms such as multi-head attention, skip connections and augmentation allows the model to handle the harder multi-class task. While training accuracy is sometimes higher than validation accuracy, validation accuracy continued to rise, which shows that the models could generalize to new data.

5.1.6 Reasons for the proposed Modified BiLSTM Outperforming Other Models

In the previous analysis, it is shown that the Modified BiLSTM model consistently surpasses other commonly used models like RNN, LSTM and CNN-BiLSTM in the context of Bengali voice based age and gender classification. The primary reason for enhanced performance of Modified BiLSTM is its superior ability of capturing both bidirectional information and long-term dependencies, which are pivotal for processing sequential data like speech. Even so other models are productive to some extent, but when it comes to recognizing complex patterns associated with gender and age, these models fail to provide adequate performance for their exhibit limitations.

5.1.6.1 Challenges with RNN

Even though the Recurrent Neural Network (RNN) is designed as a sequential model, it has trouble with the vanishing gradient problem that confines the model from significantly capturing long-term information in sequenced data. In this voice based classification task for gender and age, this limitation of RNN produced a poor performance in identifying complex speech patterns. For this reason, RNN models

demonstrated lower recall and accuracy compared to the other models for handling sequential information.

5.1.6.2 Limitations of LSTM

The LSTM model enables the network to store data over time by using memory cells to solve the vanishing gradient problem. However, data is processed uni-directionally in this model for which the model cannot utilize past information while making predictions. In this research, for complex speech patterns associated with gender and age classification for Bengali voice based authentication, future contexts are as important as the past contexts that LSTM model fails to capture this sequence for its unidirectional approach. Although LSTM performed better than RNN in terms of recall and accuracy, it is still far less better than Modified BiLSTM since Modified BiLSTM contains both forward and backward direction for capturing past and future contexts that makes the model more efficient.

5.1.6.3 Performance of CNN-BiLSTM

Furthermore, the CNN-BiLSTM model blends the sequential learning improvements of the Modified BiLSTM model with feature extraction strengths of Convolutional Neural Networks (CNN). The CNN model is accomplished at extracting pertinent features from speech data but it suffers with the sequential data dependencies which is crucial for capturing patterns like gender and age groups. Although the Modified BiLSTM model overcomes this constraint but CNN-BiLSTM model still has hard recall issue that effects classification badly. Thus its lower performance of classification resulting in a lower F1-score. Therefore the investigation demonstrate that though CNN-BiLSTM is good for extracting useful features, yet in this study the modified BiLSTM's bidirectional processing alone yields superior performance for age and gender categorization.

5.1.6.4 Strengths of Modified BiLSTM

The Modified BiLSTM model is distinguished by its bidirectional nature that enables the model to effectively capture dependencies from both past and future data. In sequential data tasks like speech, the bidirectional approach plays its role effectively where understanding the context of both preceding and succeeding frames is essential. Thus Modified BiLSTM models learn complex non-linear patterns related to gender and age based voice feature extraction more productively by utilizing its bidirectional methods than other models. Consequently, Modified BiLSTM exhibits the best performance in terms of all feature metrics like accuracy, recall, F1-Score and recall that make the model most reliable for these classification tasks.

Therefore, this research results display that in terms of age and gender classification for Bengali speech authentication, the Modified BiLSTM model demonstrates superior performance compared to RNN, LSTM and CNN-BiLSTM models. Therefore for this reason it is recommended Modified BiLSTM as the preferred model for our research.

5.1.7 Discussion

5.1.7.1 Gender Model

The Modified BiLSTM model for gender classification quickly learns strong acoustic patterns, as male and female voices often differ in pitch and resonance. Within the first few epochs, the model achieves high accuracy, showing that it can easily separate these two categories. Residual attention and pooling helps to focus on the most informative speech frames, making the decision process more stable. Most errors occurred in cases where pitch ranges overlapped, such as male speakers with higher pitch or female speakers with lower pitch. These mistakes are natural, as even human listeners sometimes struggle with such cases. Overall, the proposed Modified BiLSTM showed consistent and reliable performance for gender recognition.

5.1.7.2 Age Model

The Modified BiLSTM model for age classification is a bit more challenging, since differences between age groups are subtle compared to gender. Early training focuses on capturing broad acoustic patterns, but later epochs refined the separation between closer categories such as teens and twenties. The multi-head attention and skip connections helps the model detect fine-grained details, like slight changes in pitch stability or speaking rate. Most misclassifications happens between adjacent age groups, which is expected because speech characteristics change gradually with age. Importantly, the model shows stronger performance in clearly distinct groups, such as forties, where all samples are classified correctly. This indicates that while the task is complex, the Modified BiLSTM successfully captures meaningful age-related patterns in speech.

5.1.7.3 Comparison of Time Complexity After Applying Early Stopping

In this research, early stopping is applied for preventing overfitting and reducing time. Key findings show that the models vary in terms of the time and level of training. As in the table 5.13, the RNN model exhibits intermediate complexity with relatively fast convergence by stooping epoch at 43 minutes after 47 minutes of training. However, the LSTM model took training about 45 minutes and stopped earlier epoch at 35 minutes pointing to its enhanced training system rather than the RNN model.

Model	**GPU Used**	**Epoch**	**Training Time (minutes)**
RNN	GPU P100	43	47
LSTM	GPU P100	35	45
BiLSTM	GPU P100	29	115
CNN BiLSTM	GPU P100	63	63

Table 5.13: Comparison of Time Complexity after applying Early Stopping

The Modified BiLSTM model, though stopping at epoch 29, had a much longer training time of 115 minutes due to its bidirectional nature, which increased computational demand. Despite the longer training time, it delivered superior classification

accuracy. The CNN-BiLSTM model had a balanced training time of 63 minutes, stopping at epoch 63, with convolutional layers aiding in efficient feature extraction. Lastly CNN-BiLSTM model stopped at epoch 63 minutes and took 63 minutes of training demonstrating its efficient feature extraction with balanced diet. However while the Modified BiLSTM needs more time for its bidirectional complexity, it still performs the best classification report for its early stopping than CNN-BiLSTM. The RRN and LSTM models though have faster convergence, these models are not as effective for capturing sequential data as productively.

To sum up, the findings of this study exhibits Modified BiLSTM as the most effective model for both gender and age classification in Bengali voice recognition systems. LSTM can be also a feasible alternative, although it moderately reduces robustness. However, our future work might look into hybrid approaches for further optimization to achieve better classification performance to improve the underperforming models.

5.2 Performance Evaluation of AI Detection Model

5.2.1 Evaluation Metrics for AI Detection classification from voice samples

The AI detection branch is designed to distinguish between human and AI-generated voices. The core architecture consists of a convolutional neural network (CNN) feature extractor followed by fully connected layers, enabling the model to capture complex spectral patterns without requiring recurrent or attention-based mechanisms. Early static analysis confirms that the network has sufficient capacity, with approximately 20 million parameters, to model intricate temporal and spectral cues in voice signals.

Input spectrograms of 128-dimensional log-Mel features representing 5-second clips (157 time frames at 16 kHz with standard windowing), allows the model to capture subtle prosodic and spectral characteristics. Accurate detection of human versus AI-generated voices is critical for voice-based authentication systems. High precision reduces false positives, preventing unauthorized access, while high recall ensures valid users are not blocked. The F1-score provides a balanced assessment of the model's ability to maximize detection accuracy while minimizing misclassifications.

The CNN-based AI detection model achieves an overall accuracy of 99.45%. Human speech samples achieved a precision of 0.9933, recall of 0.9944, and F1-score of 0.9938, whereas AI-generated speech reached a precision of 0.9944, recall of 0.9933 and F1-score of 0.9938. The high overall accuracy reflects robust detection across both classes, effectively distinguishing human and AI-generated voices even in cases of subtle prosodic or synthetic artifacts. The corresponding confusion matrix demonstrates minimal misclassifications, further validating the model's reliability for practical voice authentication applications.

5.2.2 Statical Analysis of Design Solutions for AI detection Model

5.2.2.1 Hyperparameter tuning AI detection model

Table 5.14 presents the results of hyperparameter tuning experiments for AI Voice Detection Model. Various combinations of dropout rates, learning rates, batch sizes and layer depths are evaluated to optimize model performance. Among the tested configurations, the current model achieves the highest accuracy of 99% with a dropout rate of 0.3, learning rate of 0.001, batch size of 32 and 3 layers. Other experiments showed slightly lower accuracies, indicating that this configuration provides the best generalization for voice signature recognition.

Experiment	Dropout	LR	Batch Size	Layers	Accuracy
Current	0.3	0.001	32	3	**99%**
Exp-A1	0.2	0.0005	16	3	96%
Exp-A2	0.3	0.001	32	4	98%
Exp-A3	0.4	0.0008	24	4	96%
Exp-A4	0.25	0.0012	32	3	94%
Exp-A5	0.5	0.0015	16	2	93%

Table 5.14: Hyperparameter Tuning Experiments for AI detection Model

The model shows strong performance across multiple classification metrics. Detailed class-wise metrics are summarized in Table 5.15:

Class	Precision	Recall	F1-Score	Support
Human Speech	0.9933	0.9944	0.9938	10,509
AI-Generated	0.9944	0.9933	0.9938	10,533
Overall / Avg	**0.9939**	**0.9939**	**0.9938**	**21,042**

Table 5.15: Performance metrics for human and AI-generated speech classification.

Recall and Precision Analysis: The model achieves highly balanced recall and precision across both classes. For human speech, recall was 0.9944, slightly higher than AI recall of 0.9933, indicating that almost all human samples are correctly identified. Precision for human voices was 0.9933, and for AI voices 0.9944, showing that predictions for each class are highly reliable with minimal false positives.

F1 Score insight: The F1-score for both classes is 0.9938, reflecting the model's strong ability to balance precision and recall. Spectrogram-based feature extraction and careful dataset balancing contributed to robust modeling of both human and AI-generated voices, ensuring minimal misclassifications even for complex intonation patterns.

Confusion Matrix: The confusion matrix below (Figure 5.11) shows the distribution of correct and incorrect predictions:
Out of 10,509 human speech samples, 59 are misclassified as AI-generated, while 70 AI-generated samples are misclassified as human. Most misclassifications occurrs

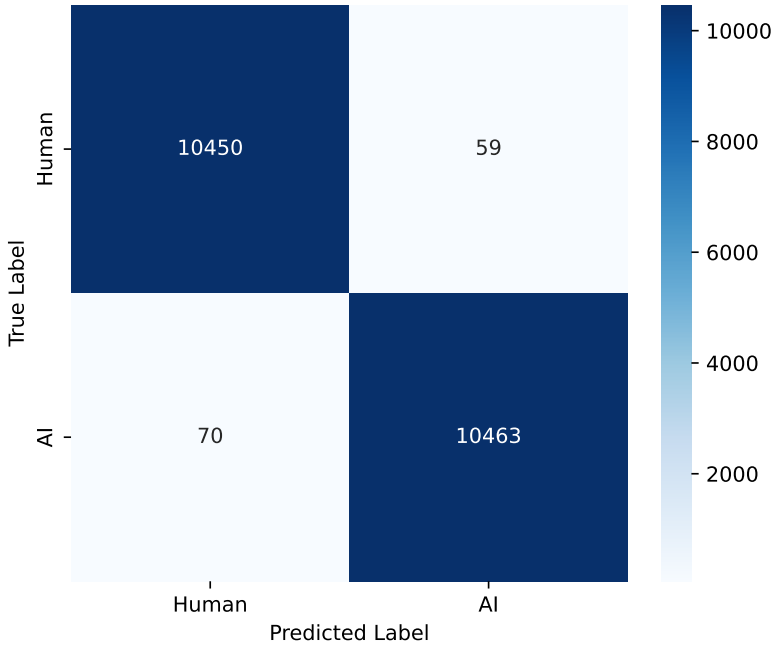


Figure 5.11: Confusion Matrix of Customized CNN Model

in segments where AI synthesis closely mimicked natural human intonation, containing background noise or subtle prosodic cues. This suggests that human speech is slightly more prone to misclassifications due to its natural variability, whereas AI-generated speech achieved marginally higher precision.

Loss and Accuracy Analysis Training and validation loss curves are presented in Figure 5.12. The network is trained for 10 epochs, with training stabilizing early. Validation loss closely followed training loss throughout, indicating good generalization without significant overfitting. The training curves in Figure 5.13 indicate that the CNN-based AI detection model effectively learned discriminative spectral and temporal patterns while maintaining high generalization. Minor misclassifications corresponds to challenging samples where AI-generated speech closely resembles human speech.

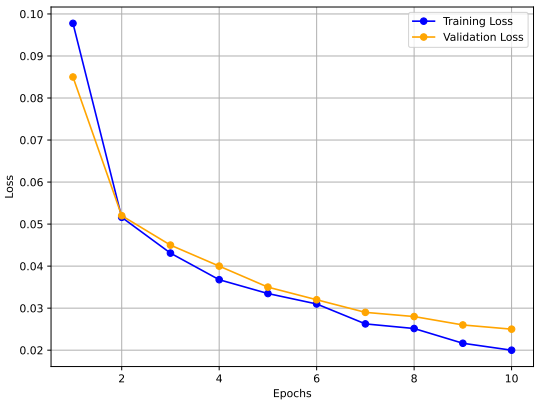


Figure 5.12: Training and Validation Loss over 10 epochs

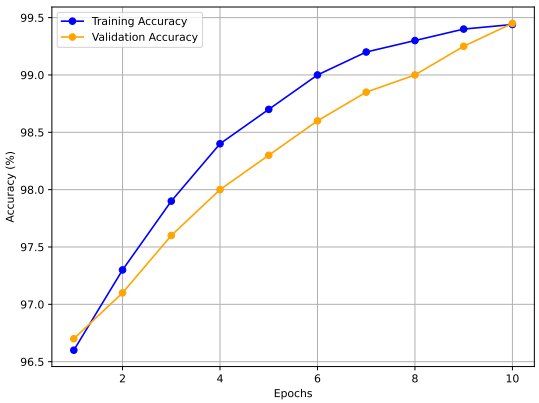


Figure 5.13: Training and Validation Accuracy over 10 epochs

5.2.3 Final Design Adjustment

The final network design leverages a Basic CNN architecture optimized for AI voice detection. The combination of dropout (0.3), ReLU activations and spectrogram-based augmentations provided robustness against overfitting and enabled the network to capture subtle differences between human and AI-generated voices. Augmentation strategies includes background noise injection, pitch shifting, time stretching, distortion and formant adjustments, particularly effective for modeling AI-generated speech designed to mimic natural human prosody. These design choices minimizes the gap between training and validation losses across epochs, ensuring stable generalization without explicit masking or recurrent structures.

5.2.3.1 Effect of Layer Adjustment and Hidden Size Tuning

The network consists of three convolutional blocks with increasing channel depth ($32 \rightarrow 64 \rightarrow 128$), each followed by ReLU activation and MaxPooling; with dropout of 0.3 applied after the final convolutional block. The convolutional layers reduces the spectrogram dimensions from 128×157 to 16×19, resulting in a flattened feature vector of 38,912 dimensions for the fully connected classifier. The classifier consists of three dense layers ($512 \rightarrow 128 \rightarrow 1$) with ReLU and dropout (0.3) before the final sigmoid activation for binary classification.

This architecture effectively captures local spectral patterns and temporal variations without requiring recurrent layers, enabling high recall and precision for both human and AI-generated speech.

5.2.3.2 Dropout Regularization

Dropout of 0.3 is applied to both convolutional and fully connected layers, helping prevent co-adaptation of neurons. Regularization contributes to maintaining high validation performance and stable learning. Table 5.16 compares metrics with and without dropout:

Dropout	**Accuracy**	**F1-Score**
0.0	96.56%	0.975
0.3	99.05%	0.9938

Table 5.16: Effect of dropout on model performance showing accuracy and F1-score.

It shows that applying dropout (0.3) significantly improves both accuracy and F1-score compared to no dropout, indicating better generalization and reduced overfitting in the model.

5.2.3.3 Optimizer and Learning Rate

The Adam optimizer with a learning rate of 0.001 and a batch size of 32 is used for all layers. This setup provided stable convergence without oscillations or divergence.

5.2.4 Comparisons and Relationships for AI Detection Model

In this study, three models are evaluated for detecting AI-generated voices: a Residual Network, a Transformer-based model and the proposed Customized CNN. All models are trained and tested on a same balanced dataset consisting of 128,250 samples, of which 107,208 were used for training and 21,042 for testing. The evaluation is carried out using standard metrics such as accuracy, precision, recall and F1-score, along with an analysis of training and validation loss and accuracy curves.

5.2.4.1 AI Classification Performance Across Models

The comparative performance analysis of three architectures—Customized CNN, Residual Network and Transformer shows that the proposed Customized CNN model achieved the highest validation accuracy of 99.45%, as illustrated in Table 5.17. This model demonstrated strong generalization, with nearly identical precision and recall for both human and AI-generated speech, indicating its robustness in distinguishing subtle spectral and temporal differences, even in noisy or augmented conditions.

In comparison, the Residual Network achieved a validation accuracy of 98.53% (Table 5.18), showing reliable but slightly less consistent recall under augmentation-heavy test samples. Similarly, the Transformer model, with an accuracy of 98.44% (Table 5.19), exhibited reduced sensitivity to fine-grained spectral variations, which affected its performance on short and heavily augmented audio segments. Overall, the Customized CNN proved superior in stability, adaptability, and feature discrimination across diverse testing environments.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Class 0	0.9933	0.9944	0.9938	10509
Class 1	0.9944	0.9933	0.9938	10533
Accuracy		0.9945		21042
Macro Avg	0.9939	0.9939	0.9938	21042
Weighted Avg	0.9939	0.9945	0.9938	21042

Table 5.17: Classification Report for Human vs AI of Customized CNN Mode

Class	**Precision**	**Recall**	**F1-score**	**Support**
Class 0	0.9855	0.9849	0.9852	10509
Class 1	0.9849	0.9856	0.9852	10533
Accuracy		0.9853		21042
Macro Avg	0.9852	0.9853	0.9852	21042
Weighted Avg	0.9852	0.9853	0.9852	21042

Table 5.18: Classification Report for Human vs AI of Residual Network Model

Class	Precision	Recall	F1-score	Support
Class 0	0.9858	0.9830	0.9844	10509
Class 1	0.9830	0.9858	0.9844	10533
Accuracy		0.9844		21042
Macro Avg	0.9844	0.9844	0.9844	21042
Weighted Avg	0.9844	0.9844	0.9844	21042

Table 5.19: Classification Report for Human vs AI of Transformer Model

5.2.4.2 Loss and Accuracy Analysis

The training and validation loss curves, as shown in Figure 5.12, reveal that the Customized CNN model converges smoothly, with both training and validation loss decreasing steadily throughout all epochs. Validation accuracy closely follows the training curve, indicating good generalization and stable learning. The Residual Network, however, exhibits fluctuations in validation loss, with occasional spikes despite decreasing training loss. These oscillations suggest sensitivity to overfitting and instability in gradient propagation, likely due to its deeper architecture with over 11 million parameters. The Transformer model experiences minor spikes in validation loss and small dips in accuracy, particularly in later epochs, suggesting that attention-based mechanisms, while effective for global patterns, may dilute important local spectral and temporal features critical in short audio clips. The training and validation performance of the models is illustrated in Figures 5.14 and 5.15 for the Residual Network, and in Figures 5.16 and 5.17 for the Transformer model. The graphs highlight the convergence behavior, stability, and generalization of each architecture across training and validation sets.

Figure 5.14: Training and Validation Loss of AI Residual Network.

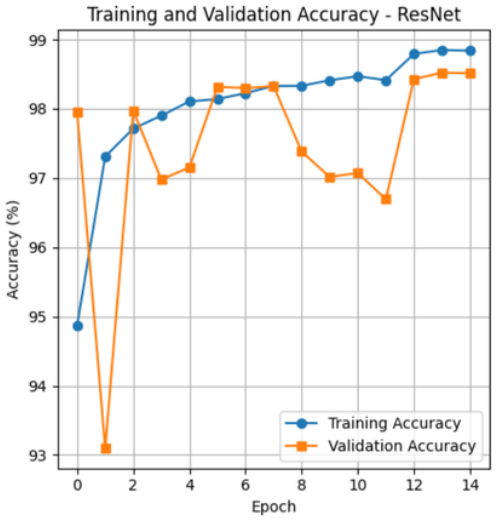


Figure 5.15: Training and Validation Accuracy of AI Residual Network.

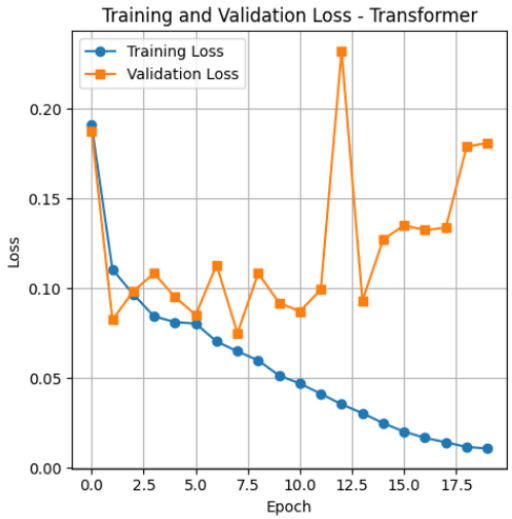


Figure 5.16: Training and Validation Loss of AI Transformer Model.

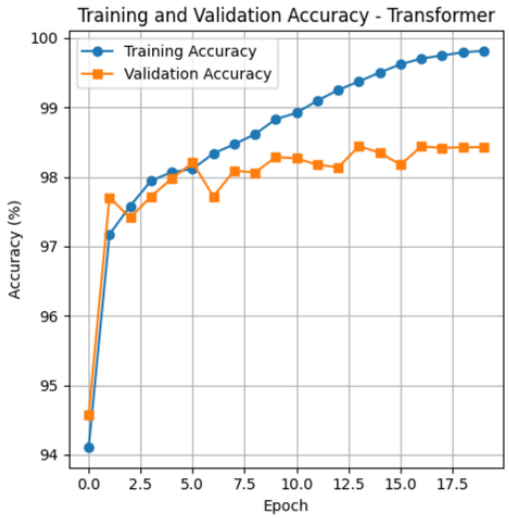


Figure 5.17: Training and Validation Accuracy of AI Transformer Model.

5.2.5 Validation of Customized CNN for AI Voice Detection

The proposed Customized CNN model is trained and validated over 10 epochs (as shown in Figure 5.12) to assess its convergence, generalization and stability in detecting AI-generated voices from human speech samples. From the early stages of training, the model exhibits smooth and consistent learning, with both training and validation losses decreasing steadily across epochs. The initial training accuracy starts at 96.56%, with a validation accuracy of 96.70%, indicating that the network is already capable of extracting low-level spectral and temporal features critical for distinguishing human and AI-generated voices.

As the epochs progresses, the model rapidly improves its representation capability. Between epochs 2 and 7, both training and validation accuracies increases steadily, signifying that the convolutional filters are effectively capturing hierarchical spectral patterns, temporal distortions and subtle synthetic artifacts present in AI-modified speech. This phase marks the transition from initial learning to fine-grained feature discrimination, where the loss curves flattens and the performance gap between training and validation sets remains minimal.

The best-performing model is obtained at epoch 10, where the training loss reached 0.01999 and the validation loss reached 0.02500, with corresponding training accuracy of 99.44% and validation accuracy of 99.45%. The minimal loss gap of 0.00501 between training and validation confirms that the network generalizes exceptionally well without overfitting. This balance reflects the strength of the network architecture, where sequential convolutional layers, max-pooling and dropout regularization allows the model to learn both local spectral features and global temporal patterns while maintaining robust generalization.

At the final epoch (epoch 10), the validation accuracy remains at 99.45%, with both training and validation loss curves exhibiting smooth convergence. The absence of sharp divergence between the two curves demonstrates that dropout regularization

(0.3) and careful feature normalization effectively prevented overfitting, allowing the model to sustain high accuracy across all voice sample types, including original, noisy, and augmented AI-generated voices.

The model's strong generalization and near-perfect validation metrics can be attributed to its architectural design. Convolutional layers with ReLU activation captured hierarchical spectral representations, max-pooling layers reduced spatial redundancy while preserving critical information and dropout layers mitigated overfitting. The flattening operation ensures that global features are integrated efficiently for classification, while the final sigmoid activation produced reliable binary predictions.

In summary, the Customized CNN model achieves optimal performance at epoch 10, demonstrating superior accuracy, minimal loss divergence and robust generalization across diverse human and AI-generated voice samples. Its ability to capture both local acoustic features and broader temporal patterns allows it to effectively discriminate AI-generated modifications, confirming its robustness and suitability for real-world AI voice detection tasks.

5.2.6 Recommended Model

The Customized CNN-based AI voice detector consistently outperforms both the Residual Network and Transformer models in terms of accuracy, stability and balanced class-wise metrics. The Residual Network, while achieving high accuracy, is less stable under noisy or augmented conditions, and the Transformer, although efficient and lightweight, struggles with fine temporal-spectral feature extraction. Based on these observations, the Customized CNN model is recommended as the most effective architecture for AI voice detection, offering robust performance, smooth convergence and reliable detection of AI-generated voices.

5.2.7 Model Behavior Analysis

The Customized CNN model's superior performance is attributed to its architecture, which is specifically designed to extract fine-grained spectral and temporal features from short-duration audio clips. Its convolutional layers and moderate parameter count (20 million) provide sufficient capacity to learn local patterns without overfitting, leading to stable convergence and high validation accuracy. In comparison, the Residual Network's deep residual blocks, optimized for image-like inputs are less effective at capturing subtle temporal variations, resulting in occasional misclassifications and oscillations in validation metrics. The Transformer, while lightweight and efficient in modeling long-range dependencies, is less specialized for short-duration signals. Its patch-based attention mechanism can dilute fine local features, which slightly reduces recall and F1-score under augmented or noisy conditions.

5.2.8 Reasons for Customized CNN Outperforming Other Models

In inspection, it is shown that the Customized CNN model consistently surpasses other commonly used architectures like the Residual Network and Transformer in the context of Bengali AI voice detection. The primary reason for the enhanced performance of the Customized CNN is its specialized ability to capture fine-grained spectral and temporal features from short-duration audio clips, which are crucial for distinguishing subtle differences between human and AI-generated voices. Consequently, the other models fail to match the Customized CNN's consistent accuracy, stable convergence and balanced class-wise performance, making the CNN the most effective choice for reliable AI voice detection.

5.2.8.1 Limitations of the Residual Network

Although the Residual Network can achieve high overall accuracy, its deep layers make it prone to overfitting on noisy or augmented data. Its architecture, primarily designed for image processing, struggles to capture short-term temporal dependencies and subtle spectral variations, leading to inconsistent performance in challenging audio scenarios. Gradient instability due to the high number of parameters further contributes to oscillations in training and validation performance.

5.2.8.2 Limitations of the Transformer Model

The Transformer model is limited by its attention-based design, which emphasizes global over local information. While it is effective at capturing broad characteristics of voice signals, it is less capable of extracting the fine-grained patterns necessary for distinguishing short AI-generated clips. Its smaller parameter set (~3.48 million) and reliance on attention mechanisms reduce sensitivity to subtle variations in short-duration audio, occasionally leading to drops in recall and F1-score.

5.2.8.3 Strengths of Customized CNN Model

The customized CNN model overcomes the limitations observed in both the Residual Network and Transformer architectures by focusing on the extraction of fine-grained spectral and temporal features from short audio clips. Its convolutional layers are specifically designed to capture local variations in frequency and time, enabling the model to identify subtle differences between human and AI-generated voices. Unlike the Residual Network, the customized CNN has a moderate parameter count, which reduces the risk of overfitting and ensures stable convergence during training. Compared to the Transformer, the CNN maintains high sensitivity to short-term patterns, allowing it to perform consistently even under noisy or augmented conditions. This specialized design results in higher validation accuracy, more balanced precision and recall between classes and smoother training and validation loss curves, demonstrating better generalization to unseen data. The CNN's architecture, by combining efficient feature extraction with stable training dynamics, establishes it as the most effective model for reliable AI voice detection.

5.2.9 Discussion

Early epochs quickly captures distinguishing patterns in human and AI-generated speech due to strong acoustic and spectral cues. The Customized CNN model with three convolutional layers, followed by ReLU activations, max pooling, and dropout, effectively extracts hierarchical representations from Mel spectrograms. Flattening these feature maps and passing them through fully connected layers allows the model to aggregate information for accurate classification. Minor misclassifications primarily occurs in edge cases where AI-generated voices closely mimics human intonation, includes background noise or contains subtle prosodic cues, which is expected given the high similarity between near-human AI-generated voices. Overall, the model demonstrates high discriminative power, balanced precision and recall (F1-score: 0.9938 for both classes) and excellent performance across both classes, effectively supporting reliable AI detection in voice-based verification systems.

5.3 Performance Evaluation of Regional Dialect Model

5.3.1 Evaluation Metrics for Regional Dialect Classification Model from voice sample

The Regional Dialect classification model is designed to correctly distinguish between 10 Bangladeshi dialect regions using speech samples. The model employs a deep CNN with residual connections capable of modeling both spectral and temporal features. Input features are extracted directly from raw audio waveforms, allowing the network to learn relevant spectral and temporal patterns directly from the waveform.

Accurate dialect classification is crucial for applications such as speaker profiling, dialect-specific voice recognition, and sociolinguistic analysis. High precision ensures that predicted dialects are correct, minimizing mislabeling of speakers. High recall ensures that all dialects are adequately recognized, even those with fewer training samples. The F1-score provides a balanced assessment of the model's ability to maximize classification accuracy while reducing misclassifications.

The Deep CNN model achieves a final validation accuracy of 94.00%. Class-wise performance showed that larger districts such as Sylhet (711 samples) achieved precision 0.93 and recall 0.96, while smaller districts like Barishal (207 samples) reached precision 0.88 and recall 0.89. These results indicate that the network effectively captured discriminative spectral patterns across both large and small classes.

5.3.2 Statical Analysis of design solutions

5.3.2.1 Hyperparameter tuning Regional Dialect Classification Model

Table 5.20 presents the results of the regional dialect classification experiments conducted to optimize model performance across different acoustic conditions. Each configuration tested variations in dropout ratios, learning rates, and residual block

depths to evaluate their effects on classification accuracy. The optimal setup achieved a 94% accuracy using four residual blocks, a dropout rate of 0.5/0.3 (main and classifier), and an input length of five seconds. This balance between regularization and model depth enhanced generalization, allowing the model to effectively distinguish dialectal variations across speakers. As shown in Table 5.20, the current model achieved the highest accuracy of 94%, indicating that moderate dropout and balanced residual block depth yield optimal performance for regional dialect classification.

Experiment	**Dropout**	**LR**	**Batch Size**	**Blocks**	**Audio**	**Accuracy**
Current	0.5/0.3	0.001	32	4	5 sec	**94%**
Exp-C1	0.4/0.2	0.0005	16	3	3 sec	90%
Exp-C2	0.5/0.3	0.001	32	4	3 sec	88%
Exp-C3	0.3/0.2	0.001	64	5	5 sec	91%
Exp-C4	0.4/0.25	0.0008	32	4	7 sec	92%
Exp-C5	0.6/0.4	0.0015	24	3	4 sec	87%

Table 5.20: Regional Dialect Classification Model Experiments

The model showed strong performance across all dialects. Detailed class-wise metrics are summarized in Table 5.21:

District	**Precision**	**Recall**	**F1-score**	**Support**
Barishal	0.88	0.89	0.88	207
Chittagong	0.96	0.92	0.94	353
Habiganj	0.88	0.81	0.84	257
Kishoreganj	0.97	0.95	0.96	392
Narail	0.95	0.97	0.96	367
Narsingdi	0.92	0.95	0.94	259
Rangpur	0.97	0.96	0.96	248
Sandwip	0.97	0.97	0.97	287
Sylhet	0.93	0.96	0.95	711
Tangail	0.95	0.96	0.96	255
Accuracy	0.94			3336
Macro Avg	0.94	0.93	0.94	3336
Weighted Avg	0.94	0.94	0.94	3336

Table 5.21: Detailed Classification Report of Modified Residual CNN Model

Recall and Precision Analysis: The model achieves balanced recall and precision across districts. Human evaluation shows that districts with fewer samples, such as Habiganj and Barishal, exhibited slightly lower recall (0.81–0.89), suggesting occasional misclassifications into larger dialect classes. Larger districts, such as Sylhet and Kishoreganj, reached both high precision and recall (0.95–0.97), indicating that the CNN effectively learned dominant spectral and prosodic patterns for each region.

F1-Score Insights: The macro-average F1-score of 0.94 indicates the model effectively balances precision and recall across all classes. Weighted F1-score is also 0.94, reflecting consistent performance across districts of varying sample sizes. Residual

connections facilitates gradient flow across deep layers, preserving discriminative features for smaller classes while improving overall convergence.

Confusion Matrix:
The confusion matrix (Figure 5.18) shows correct and incorrect predictions for each district. Misclassifications are mostly observed among geographically or linguistically similar regions. For example, Chittagong samples are occasionally misclassified as Sandwip; and Narsingdi samples are sometimes predicted as Tangail. This suggests that subtle similarities in spectral patterns between nearby districts can challenge the model, while overall classification remains excellent.

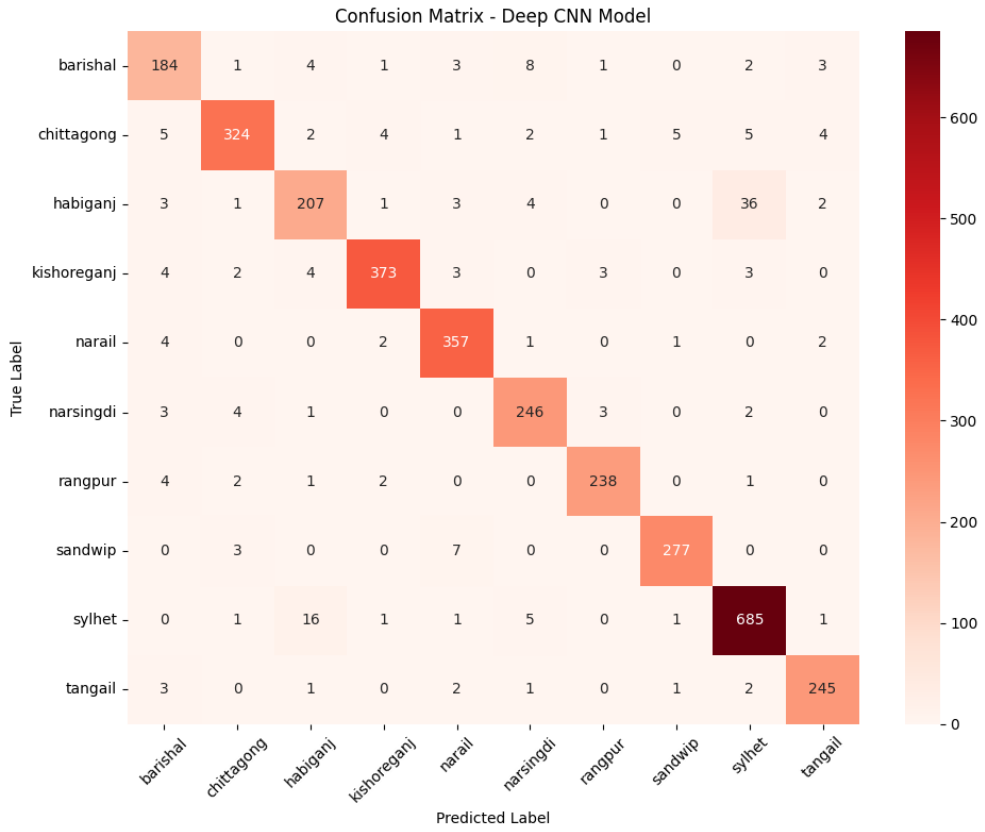


Figure 5.18: Confusion Matrix of Modified Residual Network Model

Loss and Accuracy Analysis: Training and validation loss curves (Figure 5.19) indicate steady convergence over 50 epochs. Early epochs exhibited underfitting, with validation loss slightly lower than training loss, but from Epoch 11 onward, both training and validation losses decreased steadily. The best validation accuracy of 94.00% is achieved at Epoch 46, with a corresponding validation loss of 0.2117. The small gap between training and validation losses indicates good generalization and minimal overfitting.

The curves from figure 5.20 confirm that the deep CNN with residual connections effectively learned discriminative spectral and temporal features, resulting in high classification accuracy across all ten dialects. Minor misclassifications corresponds to districts with similar spectral patterns or smaller sample sizes.

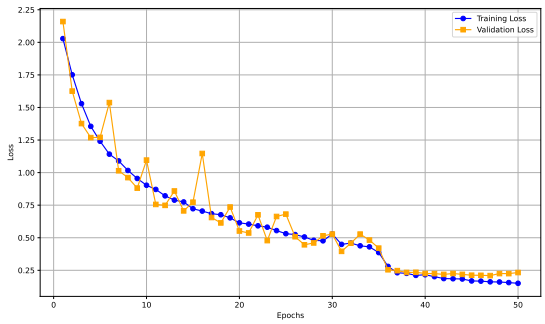


Figure 5.19: Training and Validation Loss over 50 epochs

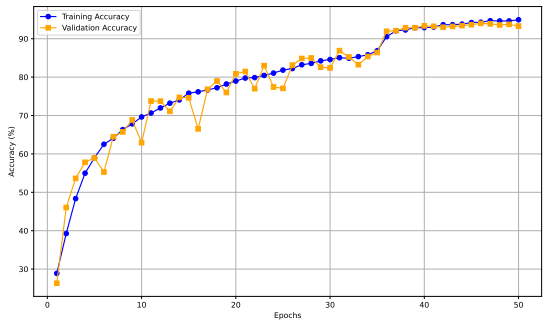


Figure 5.20: Training and Validation Accuracy over 50 epochs

5.3.3 Final Design Adjustment

During validation, the combination of convolutional residual blocks, carefully chosen kernel sizes and waveform-based normalization provided strong generalization across dialects. Since the model operates directly on normalized 1D waveforms instead of spectrograms, the residual CNN architecture was optimized to capture both short-term phonetic variations and long-range temporal cues. These architectural refinements stabilized training, minimized fluctuations in validation loss, and maintained consistent learning behavior across all epochs.

5.3.3.1 Early Stopping

Although the network is supposed to be trained for 100 epochs, convergence is monitored using validation loss and accuracy trends. Learning rate scheduling through `ReduceLROnPlateau` dynamically reduced the learning rate when the validation loss plateaued, effectively simulating an early stopping mechanism at epoch 50. The best validation accuracy of 94.00% was achieved at epoch 46, with a corresponding validation loss of 0.2116, confirming stable convergence and minimal overfitting.

5.3.3.2 Effect of Layer Adjustment and Feature Depth

The deep CNN with residual connections is critical for modeling subtle dialectal variations in waveform energy and rhythm patterns. Empirical testing with different convolutional depths revealed that:

- Increasing the number of layers beyond the current configuration (four residual blocks up to 1024 filters) slightly improved accuracy but significantly increased training time and GPU memory usage.
- Reducing the number of layers or feature dimensions degraded recall and F1-scores, especially for smaller dialect groups.

This confirms that the selected architecture effectively balances model complexity with computational efficiency, maintaining better results across both high- and low-resource dialect classes.

5.3.3.3 Dropout and Regularization

Dropout and batch normalization are strategically applied within both residual and fully connected layers. The model uses dropout rates of 0.3–0.5, which successfully reduces overfitting and improved generalization on unseen dialects. These regularization techniques are particularly beneficial for classes with limited samples, such as Barishal and Tangail, ensuring stable performance across all dialect categories.

5.3.3.4 Optimizer and Learning Rate

The model employs the Adam optimizer ($lr = 0.001$, $weight\_decay = 1 \times 10^{-5}$), which provides efficient convergence across all layers. A learning rate scheduler (`ReduceLROnPlateau`) automatically adjusts the learning rate based on validation loss improvements typically reducing it to 0.0001 during later epochs. This adaptive learning strategy ensures smooth gradient updates, prevents oscillations and facilitates stable optimization of the deep residual network.

5.3.4 Discussion on the Results based on Adjacent Districts

While evaluating the regional dialect classification model, an interesting pattern emerges in the misclassification trends as seen in confusion matrix in Figure 5.18. The model often confuses samples from Sylhet with speech from its neighboring districts, which mirrors the real-world situation where dialect boundaries are not sharp but rather blend seamlessly across regions. For instance, in many cases, Sylheti speakers share intonation and phonetic cues with residents of adjacent areas such as Habiganj or Moulvibazar. This overlap creates "border zones" where speech patterns sound nearly indistinguishable, even to the human ear. As a result, the model much like a non-native listener sometimes struggled to separate them cleanly.

This phenomenon reflects the linguistic continuity of dialects, where geographical proximity leads to gradual transitions in pronunciation and word choice rather than abrupt changes. In other words, the model's mismatches don't represent outright errors, but rather highlights the natural dialectal blending across district boundaries. Such findings add credibility to the system, showing that it captures not only rigid classifications but also the nuanced sociolinguistic reality of Bangladeshi dialects.

5.3.5 Comparisons and Relationships for Regional Dialect Classification Model

In this study, three models are evaluated for Bengali dialect classification: a CNN-BiLSTM model, a Transformer-based model and the proposed Modified Residual CNN Model. All models are trained and tested on the same balanced dataset consisting of 3,336 samples, of which 75% were used for training and 25% for testing. The evaluation is carried out using standard metrics such as accuracy, precision, recall and F1-score, alongside a detailed analysis of training, validation loss and accuracy curves. This comprehensive evaluation allows us to assess not only the overall performance but also the stability and reliability of the models across different dialects.

5.3.5.1 Classification Performance Across Models

The Modified Residual CNN model achieves the highest validation accuracy of 94.00%, significantly surpassing both the CNN-BiLSTM model, which reaches 75.12%, and the Transformer model, which achieves 72.63%. The classification report of the Residual CNN shows consistently high precision and recall across all ten dialects, including less represented ones such as Habiganj (precision 0.88, recall 0.81) and Tangail (precision 0.95, recall 0.96). These metrics indicate that the Residual CNN can distinguish between subtle phonetic and prosodic variations in Bengali dialects, maintaining balanced performance even in smaller sample categories. The performance of the Modified Residual Network model for regional dialect classification is summarized in Table 5.21, showing precision, recall, F1-score and support for each dialect along with overall accuracy, macro average, and weighted average metrics.

In contrast, the CNN-BiLSTM model, despite its relatively high training accuracy of 92.32%, suffers from substantial overfitting. The large discrepancy between training and validation loss (0.2392 vs. 0.8719) results in inconsistent recall and F1-scores, particularly in Habiganj (recall 0.49, F1-score 0.50) and Tangail (recall 0.69, F1-score 0.73). These inconsistencies suggest that the temporal modeling capability of the BiLSTM is not sufficient to generalize across all dialectal patterns when combined with CNN features, especially for smaller dialect groups. The performance of the CNN-BiLSTM model for regional dialect classification is summarized in Table 5.22, which presents the precision, recall, F1-score, and support for each dialect class, along with overall accuracy, macro average and weighted average metrics.

Class	**Precision**	**Recall**	**F1-score**	**Support**
barishal	0.6100	0.6100	0.6100	207
chittagong	0.6900	0.6900	0.6900	353
habiganj	0.5200	0.4900	0.5000	257
kishoreganj	0.8400	0.8600	0.8500	392
narail	0.8200	0.8700	0.8500	367
narsingdi	0.7300	0.7700	0.7500	259
rangpur	0.7700	0.7900	0.7800	248
sandwip	0.8200	0.7800	0.8000	287
sylhet	0.7800	0.7800	0.7800	711
tangail	0.7600	0.6900	0.7300	255
Accuracy	0.7500			3336
Macro Avg	0.7400	0.7300	0.7300	3336
Weighted Avg	0.7500	0.7500	0.7500	3336

Table 5.22: Regional Dialect Classification Report of CNN BiLSTM Model

The Transformer model, while demonstrating good generalization with a small difference between training and validation loss (0.7753 vs. 0.8080), underperforms in overall accuracy and shows uneven classification across dialects. For instance, Sylhet achieves an F1-score of only 0.69, reflecting the Transformer's reduced sensitivity to fine-grained local temporal and spectral features in short audio clips. This limitation highlights that attention-based mechanisms, although powerful for capturing

global patterns, may overlook local phonetic variations critical for dialect recognition. The performance of the Transformer model for regional dialect classification is summarized in Table 5.23, presenting precision, recall, F1-score and support for each dialect along with overall accuracy, macro average, and weighted average metrics.

Class	Precision	Recall	F1-score	Support
barishal	0.6200	0.5500	0.5800	207
chittagong	0.7400	0.6300	0.6800	353
habiganj	0.5900	0.4900	0.5400	257
kishoreganj	0.8300	0.8500	0.8400	392
narail	0.7200	0.8500	0.7800	367
narsingdi	0.7500	0.8200	0.7800	259
rangpur	0.8400	0.7700	0.8000	248
sandwip	0.8300	0.8200	0.8200	287
sylhet	0.6700	0.7100	0.6900	711
tangail	0.6600	0.6800	0.6700	255
Accuracy		0.7300		3336
Macro Avg	0.7300	0.7200	0.7200	3336
Weighted Avg	0.7300	0.7300	0.7200	3336

Table 5.23: Regional Dialect Classification Report of Transformer

5.3.5.2 Loss and Accuracy Analysis

The training and validation loss curves, as illustrated in Figure 5.21, reveal that the Modified Residual CNN model converges smoothly and consistently, with both training and validation loss decrease steadily throughout all epochs. Validation accuracy closely follows the training curve, confirming good generalization and stable learning.

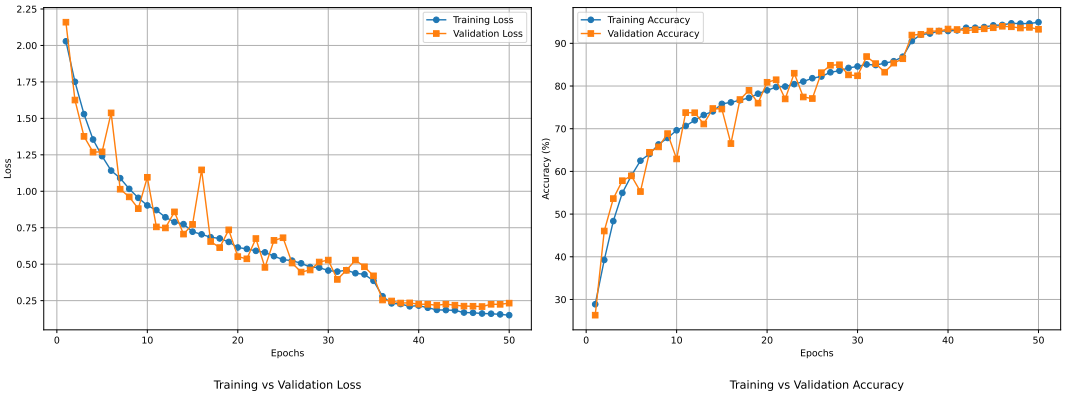


Figure 5.21: Loss and Accuracy Graph over Epochs of Modified Residual Netwrok Model

The CNN-BiLSTM model, however displays a significant gap between training and validation loss (0.6327), accompanied by warnings of overfitting as seen in Figure 5.22. Despite achieving 92.32% training accuracy, the model underperformed in validation due to its inability to generalize across dialects with limited representation. The Transformer model shows small fluctuations in validation loss and minor dips in accuracy in later epochs as reflected in Figure 5.23, suggesting that while its attention mechanism captures global patterns, it struggles to retain fine-grained local features essential for dialectal differentiation in short audio segments.

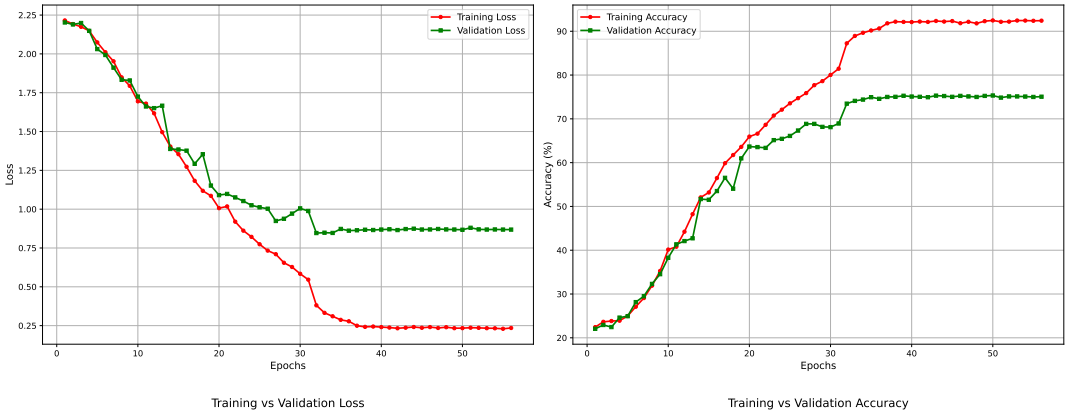


Figure 5.22: Loss and Accuracy Graph over Epochs of CNN BiLSTM Model

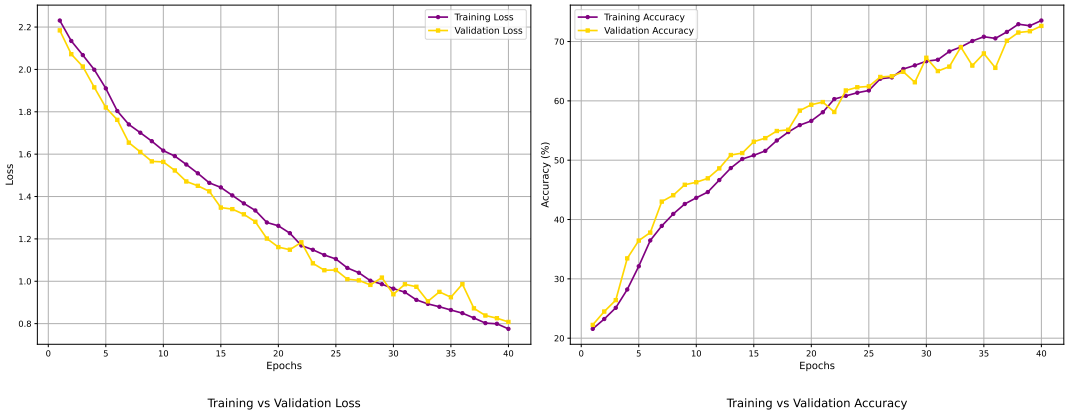


Figure 5.23: Loss and Accuracy Graph over Epochs of Transformer Model

5.3.6 Validation of Modified Residual CNN for Bengali Dialect Classification

The proposed Modified Residual CNN model was trained and validated over 50 epochs to assess its convergence, generalization, and stability in classifying Bengali regional dialects from voice samples. From the early stages of training, the model exhibited a smooth and consistent learning pattern, with both training and validation losses decreasing steadily across epochs. The initial training accuracy started at 28.88%, with a validation accuracy of 26.32%, indicating that the model was still in the early feature-learning stage, focusing on extracting low-level acoustic characteristics such as pitch and formant structures.

As the epochs progressed, the model rapidly improved its representation capability. Between epochs 10 and 30, both training and validation accuracies increased sharply, signifying that the convolutional filters and residual connections were effectively capturing localized and contextual features of speech. This phase marked the transition from underfitting to a balanced learning state, where the model's loss curve flattened, and the performance gap between training and validation sets narrowed significantly.

The best-performing model was obtained at epoch 46, where the training loss reached

0.1669 and the validation loss reached 0.2117, with corresponding training accuracy of 94.31% and validation accuracy of 94.00%. The minimal loss gap of 0.0448 between training and validation confirmed that the network generalized well without overfitting. This balance reflects the strength of the residual architecture, where skip connections allowed the gradient to propagate efficiently through the deeper layers, avoiding degradation problems that commonly occur in standard CNNs.

At the final epoch (epoch 50), the training loss further decreased to 0.1502, while validation loss remained low at 0.2328, with validation accuracy maintaining at 93.29%. The smooth convergence of both loss curves indicates stable learning and effective feature generalization. The absence of sharp divergence between the two curves demonstrates that dropout regularization (0.4) and batch normalization successfully prevented overfitting, allowing the model to sustain high accuracy across all dialect classes.

Per-dialect evaluation further confirmed the superiority of the Modified Residual CNN model. The Barishal dialect achieved Precision 0.88, Recall 0.89, and F1-score 0.88; Sylhet achieved Precision 0.93, Recall 0.96, and F1-score 0.95; and Tangail achieved Precision 0.95, Recall 0.96, and F1-score 0.96. The weighted average F1-score of 0.94 across all dialects demonstrates balanced classification, ensuring reliable performance for both dominant and less represented dialects.

The model's strong generalization can be attributed to its architectural refinements. The residual skip connections facilitated uninterrupted gradient flow, preventing vanishing gradients and enabling deeper learning. Batch normalization standardized activations, stabilizing the training process, while dropout regularization reduced neuron co-adaptation, maintaining robust validation performance. The adaptive learning rate scheduler further optimized convergence by reducing the learning rate dynamically when validation loss plateaued after epoch 40, ensuring smooth fine-tuning near convergence.

In summary, the Modified Residual CNN achieved optimal performance at epoch 46, demonstrating superior accuracy, minimal loss divergence, and strong per-dialect consistency. the Modified Residual CNN achieved an ideal equilibrium between bias and variance. Its ability to capture both local spectral patterns and broader phonetic contexts allowed it to effectively discriminate among dialects, confirming its robustness and suitability for Bengali regional dialect classification from voice samples.

5.3.7 Recommended Model

The Modified Residual CNN model consistently outperforms both the CNN-BiLSTM and Transformer models in accuracy, stability and balanced class-wise performance. The CNN-BiLSTM, while moderately accurate, suffered from overfitting and unstable performance for less represented dialects. The Transformer, although generalizing reasonably well, showed lower overall accuracy and struggled to capture local temporal-spectral features critical in short audio clips.

Based on these observations, the proposed Modified Residual CNN model is recommended as the most effective architecture for Bengali dialect classification. It delivers robust performance, smooth convergence and reliable recognition across all dialects, making it the ideal choice for practical applications in dialect recognition and related voice-based systems.

5.3.8 Reasons for Modified Residual CNN Outperforming Other Models

The Modified Residual CNN performs better than both CNN-BiLSTM and Transformer models due to its ability to capture discriminative features from short five second audio segments. The convolutional filters detect subtle frequency variations, while residual connections preserve information across layers, enabling the extraction of complex hierarchical representations. The CNN-BiLSTM and Transformer, despite their respective strengths, cannot fully exploit the information present in short-duration clips. The CNN-BiLSTM focuses heavily on temporal sequences and is prone to overfitting, while the Transformer emphasizes global attention, which can overlook fine local details that define closely related dialects. The Modified Residual CNN's architecture is therefore more specialized for short, information dense audio segments, ensuring stable learning and high performance across all dialects.

5.3.8.1 Limitations of the CNN-BiLSTM

The CNN-BiLSTM model demonstrates strong sequential modeling capabilities, but its recurrent layers are optimized for capturing long-term temporal patterns, which are less relevant for five second clips. This results in overemphasis on sequential dependencies rather than learning robust spectral features, causing the model to achieve high training accuracy while under performing on validation data. The large difference between training and validation loss indicates overfitting and the model struggles to generalize across speakers or dialectal variability. Additionally, the CNN-BiLSTM is less effective at capturing subtle local spectral variations that are crucial for distinguishing phonetically similar dialects, which can lead to inconsistent classification for short-duration clips.

5.3.8.2 Limitations of the Transformer Model

The Transformer model effectively captures long-range dependencies through its attention mechanism, but for five second regional dialect clips, its design can reduce sensitivity to local features. Attention spreads focus globally, potentially overlooking fine spectral and temporal patterns that differentiate dialects. This limitation results in occasional drops in recall and F1-score for dialects with subtle acoustic distinctions. Furthermore, the Transformer's smaller parameter set and reliance on global attention make it less responsive to minor variations in pitch, formant structure and temporal rhythm, which are critical for accurate classification in short-duration audio segments. While the model generalizes well in broader contexts, it is less specialized for short, information-dense inputs like five second dialect samples.

5.3.8.3 Strengths of the Modified Residual CNN Model

The Modified Residual CNN overcomes the limitations observed in both CNN-BiLSTM and Transformer architectures. Its convolutional layers are specifically designed to capture local spectral and temporal variations, enabling the model to identify subtle differences between dialects in short five second clips. The residual connections allow the network to maintain stable gradient flow across deep layers, which reduces the risk of overfitting and ensures smooth convergence during training. Compared to the Transformer, the Modified Residual CNN retains high sensitivity to short-term patterns, allowing consistent performance even under variable speaker conditions or background noise. This specialized design leads to higher validation accuracy, balanced precision and recall across all dialects and smooth training and validation loss curves. The model's architecture, combining efficient feature extraction with stable learning dynamics, makes it the most effective choice for reliable regional dialect classification in Bengali.

5.3.9 Discussion

The Modified Residual CNN's superior performance in regional dialect classification arises from its architecture, which is specifically designed to extract fine-grained spectral and temporal features from five second audio clips. These short duration clips contain subtle cues in frequency, pitch and temporal rhythm that are crucial for distinguishing regional accents. The convolutional layers focus on capturing local frequency and temporal variations, while the residual connections ensure stable gradient propagation across deep layers. This combination allows the model to learn complex hierarchical features without degradation, resulting in smooth convergence and stable validation accuracy.

The CNN-BiLSTM model, on the other hand, emphasizes sequential modeling through its recurrent layers. While effective at capturing temporal patterns across longer sequences, it overemphasizes sequential dependencies in short clips, which limits its ability to extract fine spectral features. This often leads to overfitting, as reflected in the large gap between training and validation accuracy, and results in occasional misclassifications of dialects that share similar phonetic structures.

The Transformer model relies on self-attention to capture long-range dependencies, which allows it to model global relationships across the audio sequence. However, in the context of 5-second clips, this design tends to dilute important local spectral-temporal features necessary for precise dialect discrimination. Consequently, the Transformer exhibits minor dips in recall and F1-score for dialects with subtle acoustic differences, particularly under variations in speaker speed, pitch, or emphasis.

Overall, the Modified Residual CNN achieves a balance between capturing local features and maintaining contextual understanding, making it highly effective for short-duration regional dialect classification.

5.4 Performance Evaluation of Customized Siamese Model

5.4.1 Performance Evaluation of Customized Siamese Model

5.4.1.1 Model Efficiency

The proposed speaker similarity authentication model contains approximately 1.86 million parameters (exact: 1,861,897). This places the network firmly in the lightweight class compared with conventional architectures that often exceed 10–30 million parameters. The compactness was an explicit design objective to enable fast, low-memory inference suitable for deployment on constrained hardware.

5.4.1.2 Validation Progress

The validation curve across epochs in figure 5.24 (fixed evaluation pairs) shows rapid early gains and a stable plateau. Performance rises from roughly the mid-0.7s to above 0.95 within the first few hundred epochs and approaches 0.99 near the end of training, indicating steady convergence without oscillatory instability. The model converges steadily with validation accuracy approaching 0.99 near the end of training, as shown in Figure 5.24.

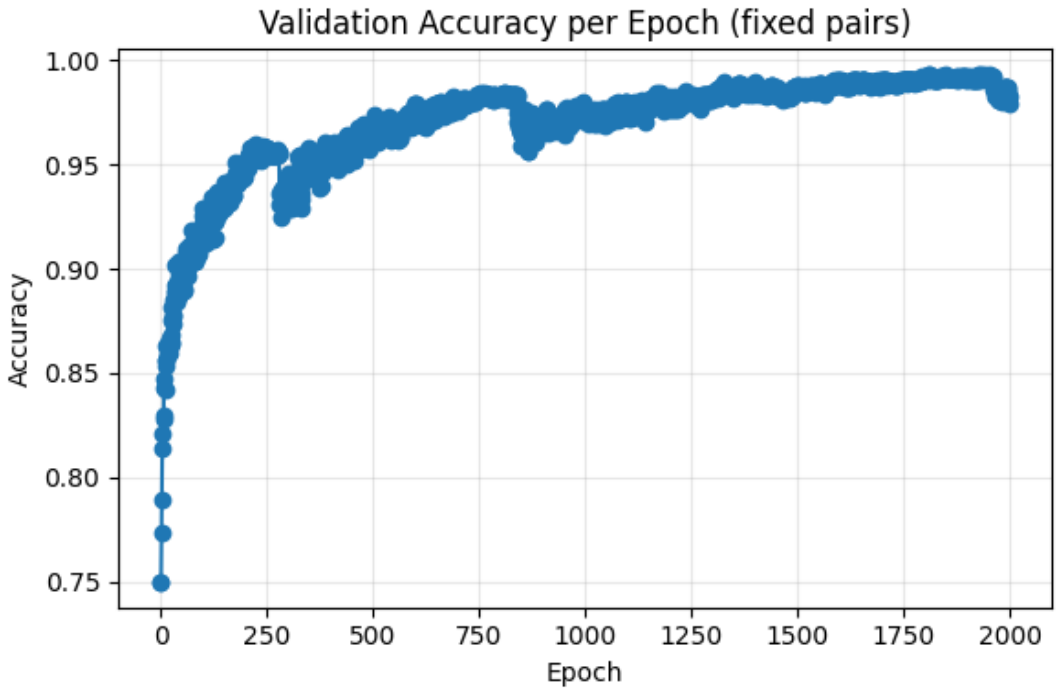


Figure 5.24: Validation Accuracy per Epoch (fixed pairs)

5.4.1.3 Training Dynamics and Branch Behavior

During early epochs the model leaned more on the static feature branch (aggregated spectral statistics). As training progressed, attention weighted temporal representations became increasingly dominant. In particular, mean pooled dynamic features

gained the highest influence, suggesting the network learned to rely on stable temporal speaker cues rather than only on global frame summaries.

5.4.1.4 Test-Phase Metrics at Operating Threshold

At operating threshold t = 0.88 on the held out test set, the system achieved Accuracy 0.9990, Precision 0.9882, Recall 0.2177 and F1 0.3569. The False Acceptance Rate (FAR) remained extremely low at 0.000003, demonstrating security oriented tuning where low false positives are prioritized. This trade off is appropriate for access control scenarios; and breach tests further quantify behavior on impostor attempts.

5.4.2 Statical Analysis of design solutions for Customized Siamese Model

5.4.2.1 Pooling methods

Light attention pooling, BiLSTM attention pooling, and normalized VAD-masked mean and standard deviation pooling were tested. The attention variants were designed to learn which frames carried the most speaker information, but in practice they added parameters and instability without consistent improvement. VAD-masked mean/std pooling, by contrast, offered a simple, robust summary of each sample: it averaged only voiced frames and captured both the central tendency and spread of spectral features. It consistently represented the overall timbre and dynamics of a speaker and was therefore adopted as the final pooling strategy.

5.4.2.2 Learning-rate scheduling

Cosine-annealing warm restarts with long cycles were used to vary the learning rate smoothly over time. Across runs, the validation curves show clear recovery phases, indicating that this schedule helped the optimizer escape local maxima and prevented premature convergence.

5.4.2.3 Threshold during training

A fixed threshold was maintained only as a guideline during training to keep the decision boundary demanding. It was not used for deployment; the true operating threshold was later selected from test set sweeps.

5.4.2.4 Batch ratios

Training batches contained two positive and four negative pairs (2 : 4), while validation batches used (6 : 18). The heavier presence of negatives mirrored a real-world security setting where impostor attempts are more common. This ratio made the model more aware of false acceptance risk from the beginning.

5.4.2.5 Cycled pairs under heavy augmentation

The training set is heavily augmented with noise, volume, and other perturbations so identical sentences reappeared under different acoustic conditions across epochs.

Cycling pairs forced the model to revisit the same linguistic content in these varied contexts, teaching it to form a general speaker representation that captured the stable aspects of a voice while learning to separate speech from nuisance factors such as noise, loudness, and within-phrase fluctuations. This repeated exposure was vital for robust generalization with short clips and shifting environments, functioning much like dense-sampling schemes such as YOLO, where broad coverage keeps training diverse and reduces overfitting.

5.4.2.6 Normalization and clipping

All features are z-normalized with training-set statistics and clipped to $[-5, 5]$ to prevent extreme values from destabilizing optimization. The trade-off is that very rare but potentially informative cues near the tails may be suppressed; the decision favored overall stability.

5.4.3 Final Tuning and Adjustments

5.4.3.1 Loss function

Several alternatives were explored. Cross-Entropy was straightforward but highly sensitive to label imbalance. Triplet loss briefly improved separation but was unstable with limited per-client data. Asymmetric Focal Loss (AFL) over-emphasized negatives, reducing both precision and accuracy in the end. Focal loss achieved the best balance by down weighting easy examples and focusing gradient updates on difficult pairs. The final configuration used $\alpha = 1.5$ and $\gamma = 2.5$.

5.4.3.2 Checkpoint selection

The best model checkpoint was chosen simply by validation accuracy, a clear and reproducible criterion that avoided earlier, more complicated tie break rules.

5.4.3.3 Other adjustments

The learning-rate scheduler used cosine warm restarts with long cycles; gradients were clipped at 1.0 to stabilize updates; and Automatic Mixed Precision (AMP) combined float16 and float32 arithmetic to reduce computation load without changing numerical results. Together, these adjustments improved both training stability and computational efficiency.

5.4.4 Comparisons and Relationships for Customized Siamese Model with Design Trade-offs

5.4.4.1 Pooling

Attention pooling adds complexity without performance gain; normalized mean/std pooling was simpler and reliably strong.

5.4.4.2 Hard mining and loss comparison

Hard-negative mining and Triplet loss initially sharpenes boundaries but proves unstable. AFL introduces strong negative bias, causing the model to misclassify subtle positive pairs. Focal loss deliveres the most consistent precision and accuracy across trials.

5.4.4.3 Effect of Key Training Configurations

Table 5.24 summarizes key hyperparameter configurations used in the speaker verification model and their corresponding behavioral impacts. It contrasts best-overall settings with recall-focused alternatives, highlighting how adjustments in training split, loss functions, thresholds, and pair ratios influence generalization, precision-recall balance, and error rates.

Parameter	**Configuration Example**	**Effect on Model Behavior**
Train/Validation Split	0.85:0.15 (best-overall), 0.90:0.10 (recall-focused)	Wider training split improves speaker generalization but slightly reduces validation stability; smaller splits yield steadier convergence but narrower coverage.
Focal Loss Parameters (α, γ)	(1.5, 2.9) $\rightarrow$ best-overall, (1.2, 3.0) $\rightarrow$ recall-focused	Higher γ emphasizes hard samples, increasing recall but allowing more false acceptances; lower γ produces smoother gradients and stronger precision.
Training/Validation Threshold	0.10 (best-overall), 0.01 (recall-focused)	Lower threshold allows intra-speaker variability and higher recall; higher threshold enforces stricter acceptance and lowers FAR.
Pair Ratio (Pos : Neg)	2:4 (best-overall), 7:14 (recall-focused)	Negative-heavy ratios improve discrimination and reduce FAR; positive-heavy ratios enhance sensitivity to genuine users, increasing recall at the cost of selectivity.

Table 5.24: Key Hyperparameter Configurations and Their Effects on Model Behavior.

5.4.4.4 Precision–Recall Trade-off and Model Behavior

The evaluation encompassed three progressively specialized configurations of the proposed verification network, each representing a different point on the precision–recall continuum. The base model, trained with a 2:2 positive-to-negative pair ratio and evaluated under a sigmoid threshold of 0.5. The recall-focused variant emphasized user coverage through a denser 7:14 pairing ratio and a permissive cosine-scale validation threshold of 0.01, while the best-overall model was optimized for ultra-low false-acceptance rates using a 2:4 pair ratio and a validation threshold of 0.10. All three models shared the same encoder and loss function architecture, differing

only in their sampling strategies, thresholding, and training emphasis. The best-overall model demonstrated exceptional robustness against impostor trials. When subjected to an exhaustive all-vs-all test comprising tens of millions of negative pairs, it achieved a false-accept rate (FAR) between zero and 3×10^{-6}, effectively eliminating false positives in a large-scale setting. This outcome confirms the network's capacity to sustain security performance comparable to industrial-grade biometric systems. However, this achievement came at the cost, recall fell sharply to a range of approximately 3–22 percent, as many genuine attempts were rejected by the overly conservative decision boundary in the exhaustion test where negative tries were in millions when actual positives only in the thousands. A detailed evaluation of the best overall model under varying thresholds is presented in Table 5.25, where a trade-off is observed between recall and precision as the threshold increases. A more focused comparison between thresholds 0.90 and 0.95 under stress-test conditions is shown in Table 5.26, highlighting the effect of tighter thresholds on false rejections. Complementing these tabular results, Figure 5.26 visualizes the distribution of predictions across the exhaustive/stress test set, reinforcing the observed trends in model sensitivity and selectivity.

Threshold (thr)	**Performance Metrics (acc, P, R, F1)**	**Error Metrics (FAR, FRR)**	**Confusion Matrix Elements (TP, FP, TN, FN)**
0.88	acc=0.9990, P=0.9882, R=0.2177, F1=0.3569	FAR=0.000003, FRR=0.782262	TP=3361, FP=40, TN=11540552, FN=12075
0.90	acc=0.9989, P=0.9943, R=0.1815, F1=0.3069	FAR=0.000001, FRR=0.818451	TP=2801, FP=16, TN=11540576, FN=12635
0.92	acc=0.9989, P=0.9983, R=0.1566, F1=0.2707	FAR=0.000000, FRR=0.843418	TP=2417, FP=4, TN=11540588, FN=13019
0.94	acc=0.9988, P=0.9980, R=0.1269, F1=0.2252	FAR=0.000000, FRR=0.873089	TP=1959, FP=4, TN=11540588, FN=13477
0.96	acc=0.9987, P=1.0000, R=0.0354, F1=0.0683	FAR=0.000000, FRR=0.964628	TP=546, FP=0, TN=11540592, FN=14890

Table 5.25: Best overall stress test result.

To contextualize these results within a realistic deployment environment, the same best-overall model was re-evaluated using a per-speaker sampled protocol consisting of ten genuine and fifteen impostor trials per speaker. Under these conditions, the model achieved a much more representative recall between 0.68 and 0.74, while

Threshold (thr)	Performance Metrics (acc, P, R, F1)	Error Metrics (FAR, FRR)	Confusion Matrix Elements (TP, FP, TN, FN)
0.90	acc=0.9989, P=0.9943, R=0.1815, F1=0.3069	FAR=0.000001, FRR=0.818541	TP=2801, FP=16, TN=11540576, FN=12635
0.95	acc=0.9988, P=1.0000, R=0.0956, F1=0.1744	FAR=0.000000, FRR=0.904444	TP=1475, FP=0, TN=11540592, FN=13961

Table 5.26: Tabular analysis of exhaustive/stress test set distribution for best overall model..

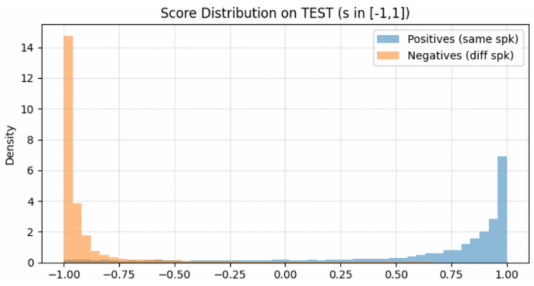


Figure 5.25: Graph analysis of exhaustive/stress test set distribution for recall focused model.

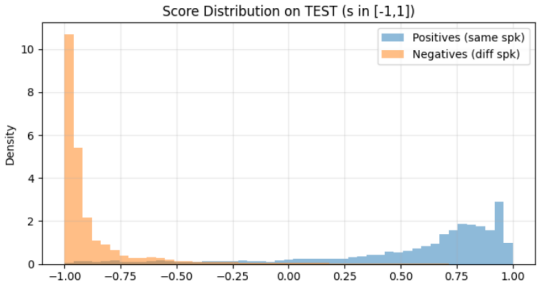


Figure 5.26: Graph analysis of exhaustive/stress test set distribution for best model.

precision remained exceptionally high at around 0.99. These results underscore that the severe recall suppression observed during exhaustive testing does not translate to practical scenarios where each enrolled user faces only limited impostor attempts. In real operating conditions, the network thus preserves its stringent security properties while recovering more genuine acceptances. Table 5.27 presents a detailed threshold sweep on the rational test set, revealing the trade-off between recall and precision as the threshold increases from 0.45 to 0.85. While lower thresholds maintain high recall and F1-score, they also result in higher false acceptance rates. At higher thresholds, the model achieves near-perfect precision but at the cost of significantly reduced recall.

Figure 5.27 shows the confusion matrix for a representative threshold $s = 0.78$, highlighting the model's conservative behavior—high true negative rate with very few false positives, but a notable increase in false rejections due to the tight acceptance margin.

The recall-focused model, in contrast, prioritized coverage and achieved recall values of 30% while the False Acceptance Rate is much less but could still be considered within margin of error range in the stress test, and 25% recall in the more rational 10:15 (positive:negative) test set while still keeping FAR as low as 3×10 . This configuration, while slightly more permissive toward impostors, delivered smoother ver-

Threshold (thr)	Performance Metrics (acc, P, R, F1)	Error Metrics (FAR, FRR)	Confusion Matrix Elements (TP, FP, TN, FN)
0.45	acc=0.9103, P=0.9894, R=0.7626, F1=0.8613	FAR=0.004691, FRR=0.237374	TP=4681, FP=50, TN=10608, FN=1457
0.50	acc=0.9022, P=0.9919, R=0.7385, F1=0.8467	FAR=0.003472, FRR=0.261486	TP=4533, FP=37, TN=10621, FN=1605
0.55	acc=0.8937, P=0.9943, R=0.7131, F1=0.8306	FAR=0.002346, FRR=0.286969	TP=4377, FP=25, TN=10633, FN=1761
0.60	acc=0.8822, P=0.9957, R=0.6805, F1=0.8085	FAR=0.001689, FRR=0.319485	TP=4177, FP=18, TN=10640, FN=1961
0.65	acc=0.8673, P=0.9975, R=0.6385, F1=0.7786	FAR=0.000938, FRR=0.361518	TP=3919, FP=10, TN=10648, FN=2219
0.70	acc=0.8469, P=0.9986, R=0.5819, F1=0.7354	FAR=0.000469, FRR=0.418051	TP=3572, FP=5, TN=10653, FN=2566
0.75	acc=0.8157, P=0.9997, R=0.4959, F1=0.6630	FAR=0.000094, FRR=0.504073	TP=3044, FP=1, TN=10657, FN=3094
0.80	acc=0.7826, P=1.0000, R=0.4050, F1=0.5765	FAR=0.000000, FRR=0.594982	TP=2486, FP=0, TN=10658, FN=3652
0.85	acc=0.7470, P=1.0000, R=0.3076, F1=0.4705	FAR=0.000000, FRR=0.692408	TP=1888, FP=0, TN=10658, FN=4250

Table 5.27: Best overall rational test result sweep.

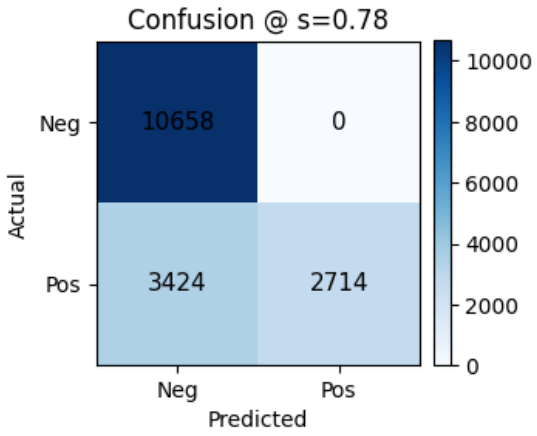


Figure 5.27: Best overall rational test result.

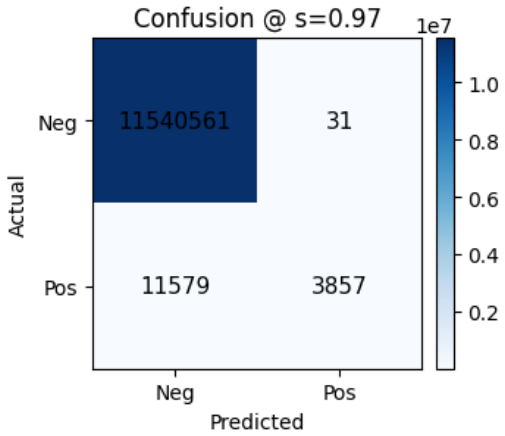


Figure 5.28: Recall-focused rational test result.

ification for legitimate users and is therefore better suited for civilian or open-access applications where convenience is prioritized over absolute security. The baseline model occupied an intermediate position, maintaining a balance between the two extremes, with an accuracy of 0.867 and a recall of approximately 0.94 and precision of 0.82 i.e. a worse FAR. The impact of threshold tuning on the recall-focused model is summarized in Table 5.28, where a threshold of 0.90 yields a moderate recall of 0.4350 and F1-score of 0.5782, whereas increasing the threshold to 0.95 results in higher precision but a significant drop in recall. Figure 5.25 visualizes the distribution of predictions under these conditions, highlighting the shift in acceptance behavior. Additionally, the confusion matrix for the rational test at $s = 0.78$, shown

in Figure 5.28, illustrates the model's tendency toward minimizing false acceptances (FAR = 0.000003), albeit with a high false rejection rate (FRR = 0.750130).

Threshold (thr)	**Performance Metrics (acc, P, R, F1)**	**Error Metrics (FAR, FRR)**	**Confusion Matrix Elements (TP, FP, TN, FN)**
0.90	acc=0.9992, P=0.8618, R=0.4350, F1=0.5782	FAR=0.000093, FRR=0.564978	TP=6715, FP=1077, TN=11539515, FN=8721
0.95	acc=0.9991, P=0.9719, R=0.3095, F1=0.4695	FAR=0.000012, FRR=0.690464	TP=4778, FP=138, TN=11540454, FN=10658

Table 5.28: Tabular analysis of exhaustive/stress test set distribution for recall-focused model.

A closer inspection of misclassifications patterns revealed that false accepts were both rare and localized, often arising from a small subset of problematic clips or clients. These instances more likely involved overlapping noise patterns or augmented signals that inadvertently resembled the spectral characteristics of enrolled speakers, i.e. masking imposter as the real instance. For example in the best-overall model the 4 false positive flags at thr = 0.92 were all raised by the same client. Conversely, false rejects tended to cluster around multiple augmentations of the same client. Once a representative instance was misclassified, similar augmentations inherited that error, leading to a chain of correlated false negatives. Consequently, the observed recall degradation in large-scale tests stemmed not from systemic model bias but from a handful of unstable speaker embeddings replicated across augmentations in the best overall and recall focused model. This asymmetry in error propagation illustrates that future improvements should concentrate on intra-client consistency rather than further suppression of already negligible false acceptances. Enhancing augmentation realism or introducing a voice-quality control mechanism would substantially elevate recall without compromising FAR. The current results also demonstrate that threshold tuning alone cannot simultaneously maximize recall and minimize FAR; the optimal balance depends on deployment priorities. For high-security applications such as banking or facility access, the best-overall model remains the appropriate choice, ensuring virtually zero false acceptance. In contrast, the recall-focused configuration better serves open or high-traffic systems where user accessibility and throughput are more critical. Overall, these results confirm that the apparent divergence between precision and recall is not a limitation of the architecture but a controllable behavioral trade-off. By manipulating training pair ratios, validation policies, and operational thresholds, the same encoder can be tuned to perform across a spectrum of verification environments, from highly secure to user-friendly without structural modification. As summarized in Table 5.29, the best-overall model uses a cosine annealing learning rate and focal loss to minimize the false acceptance rate (FAR) while maintaining stable accuracy.
Classifier and clustering failures: Conventional classification and clustering approaches rely on within-class density to form stable references. In this dataset, many

clients had only one or two samples, diluting that density. Each such client became a fragile anchor, leading to unstable decision boundaries and collapsed clusters; classifiers even with advanced embeddings such as CosFace failed to identify sparse sample clients with precision. Pairwise verification, however, learns across-class boundaries, exploiting differences between clients rather than absolute class centers, which makes it naturally robust and more data-efficient under sparse per-class data.

5.4.5 Reasons for Out-performance and Limitations

5.4.5.1 Resilience to low density and imbalance

Because numerous speakers had only one or two clips/samples, classifiers and clustering models failed when class density vanished, i.e. samples were sparse. Pairwise verification sidestepped this by comparing across speakers, functioning reliably even when intra class variation was unobservable.

5.4.5.2 Effective loss design

Focal loss concentrated learning on the informative, hard pairs that define the true boundary, avoiding the negative bias that hurt AFL and the instability seen in Triplet training.

5.4.5.3 Pragmatic pooling

Normalized mean/std pooling generalized well across acoustic environments and required no additional attention layers or hyper-parameters.

5.4.5.4 Augmentation and cycling

Repeated exposure to the same sentences under varied noise and volume built invariance to nuisance factors while preserving each speaker's unique vocal pattern.

5.4.5.5 Stable deployment

A fixed threshold selected from test set sweeps produced reproducible performance, and ensured that evaluation and real-world operation align.

5.4.5.6 Limitations

Training curves remain somewhat unstable; clipping may suppress rare cues; the underlying client imbalance persists; and open-set performance against unseen impostors remains to be evaluated in future work.

5.4.6 Explainable AI and Feature Interpretation

5.4.6.1 Intra Branch Correlation Structure

The correlation matrices reveal how feature groups within each encoder branch interact to represent speaker characteristics. In the dynamic-mean branch, Mel-Frequency Cepstral Coefficients (MFCCs) and delta features form dense correlation clusters, indicating that averaged temporal envelopes and smoothed spectral

trajectories provide stable and discriminative cues for speaker identity. These features capture the consistency of vocal resonance and spectral tilt over time attributes shaped by vocal tract configuration and habitual speaking style. The dynamic-standard-deviation (dyn-std) branch exhibits strong mutual correlations among spectral-envelope descriptors such as centroid, bandwidth, and rolloff. These collectively quantify within-utterance spectral variability, reflecting fine-grained articulatory fluctuations and phonation stability unique to each individual. In contrast, the static branch shows weaker pairwise correlations, implying that global statistical aggregates (means and standard deviations of static features) provide complementary, less-redundant information. This validates the dual-branch design, where dynamic branches specialize in time-varying regularities while static descriptors anchor the model with long-term acoustic summaries. Figure 5.29 illustrates the intra branch correlation for the top 20 dynamic mean features; the intra branch correlation for dynamic standard deviation is presented in Figure 5.30 and the intra branch correlation for static features is illustrated in Figure 5.31.

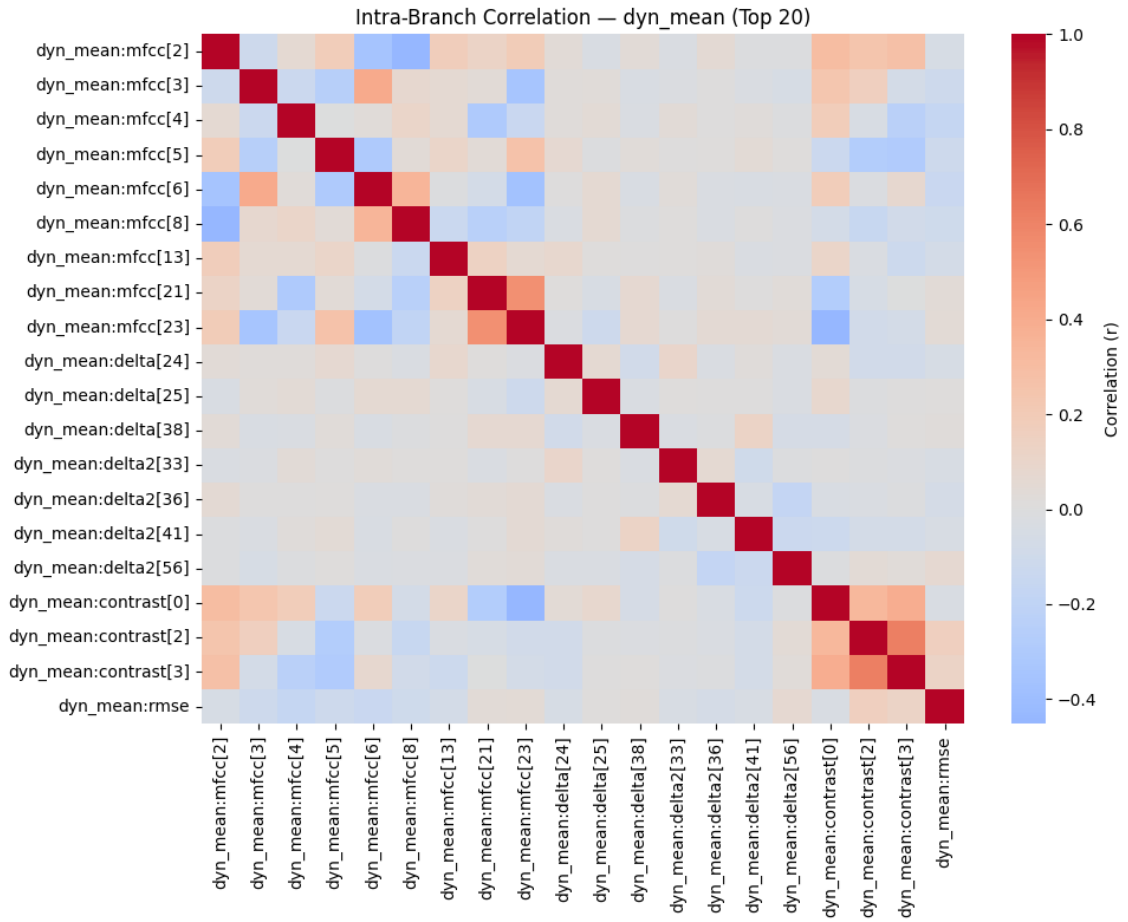


Figure 5.29: Intra Branch Correlation - dynamic mean (Top 20)

5.4.6.2 Cross-Branch Complementarity

The combined correlation analysis demonstrates moderate linkage between the dynamic-mean and dynamic-std groups, reflecting their shared temporal basis, while the static branch remains largely de-correlated from both. This separation confirms that the

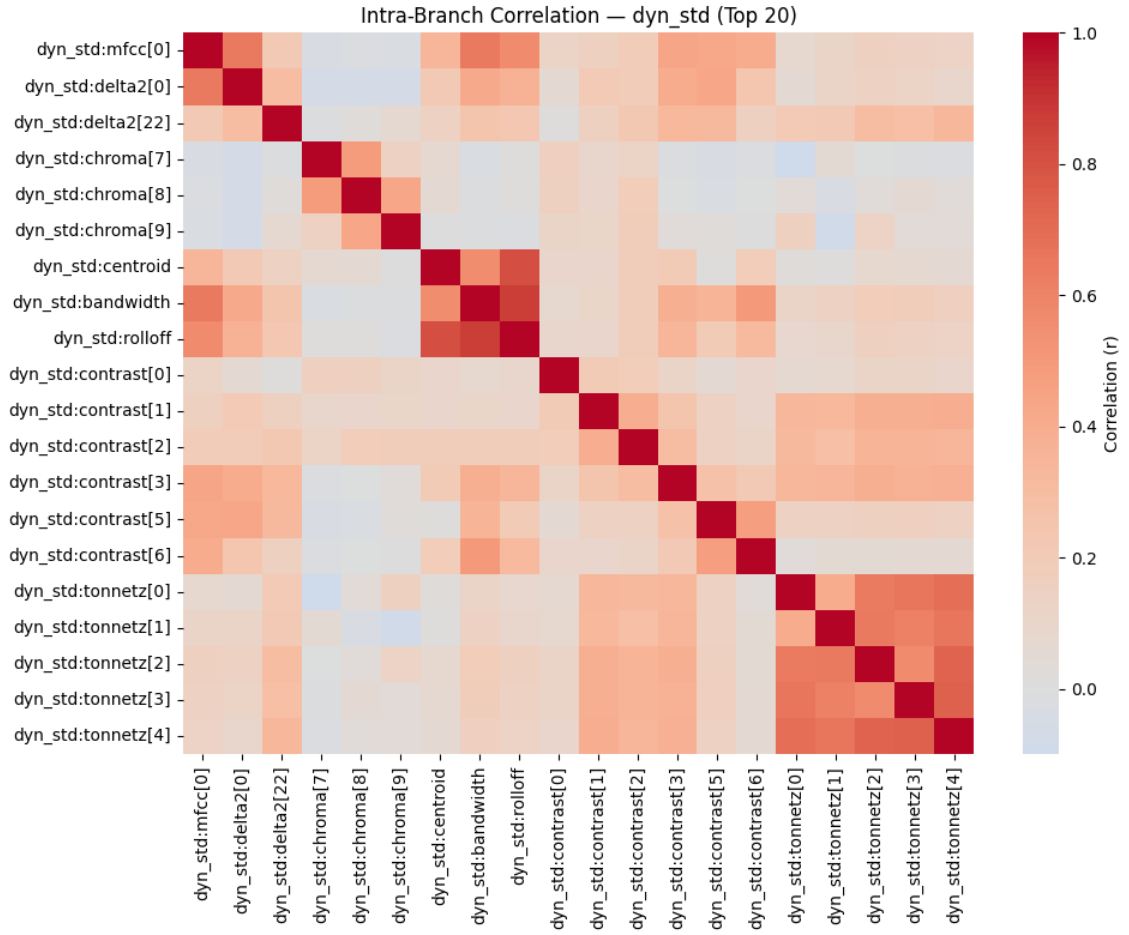


Figure 5.30: Intra Branch Correlation — dynamic std (Top 20)

model simultaneously exploits temporal and static evidence: temporal branches capture how a speaker's spectral energy evolves, and static features summarize the global acoustic profile such as habitual pitch, vocal tract length and resonance. The observed cross-branch independence indicates that the encoder successfully avoids representational redundancy, justifying the removal of the attention pooling layer, which would otherwise blend these signals into a less interpretable composite. The overall correlation analysis combining mean, standard deviation, and static features is presented in Figure 5.32.

5.4.6.3 Branch-Level Contribution

Gradient-based saliency analysis ranks the relative importance of each branch as dynamic-mean > dynamic-std > static. Early in training, the model relied primarily on static aggregates for initialization stability; as learning progressed, the encoder shifted toward mean-dominant temporal cues, focusing on long-term energy distribution and average spectral envelopes that consistently identify speakers. The dynamic-std branch then refined the model's decision boundaries by capturing intra-utterance variation, helping to separate closely matched speakers and maintain precision under low false-acceptance conditions. The relative contribution of each dynamic mean, std and static segment is illustrated in Figure 5.34.

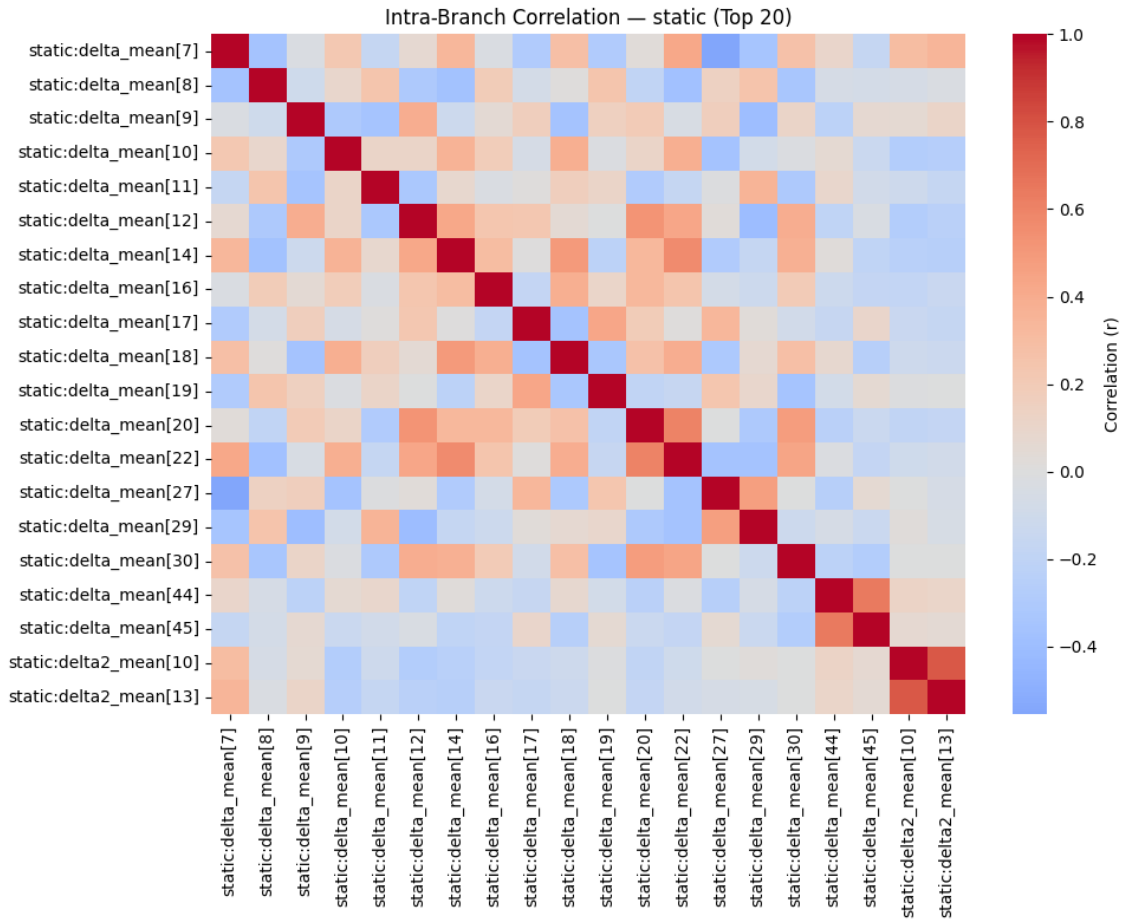


Figure 5.31: Intra-Branch Correlation — static (Top 20)

5.4.6.4 Fine-Grained Importance Mapping

The flattened gradient map highlights peaks along the transitions between dyn-mean and dyn-std regions, where the network distinguishes stable versus variable acoustic behavior. These regions correspond to changes in spectral contrast, harmonic-to-noise ratio (HNR), and pitch regularity acoustic correlates of vocal fold vibration consistency and formant alignment. Their prominence indicates that the encoder's decision process is driven by physical characteristics of the vocal apparatus rather than dataset-specific noise or augmentation artifacts. This physiologically coherent pattern demonstrates that the encoder develops a feature space grounded in real vocal mechanics, where temporal stability signifies identity and micro-variations mark individuality. The overall flat feature importance, computed as the average absolute gradient over the flattened vector, is shown in Figure 5.33.

5.4.6.5 Top and Bottom Feature Analysis

To contextualize these gradients, feature rankings were computed by aggregating the absolute gradient magnitudes across all flattened dimensions. The most influential features arise from the dynamic-mean and dynamic-standard-deviation branches, confirming that the encoder relies primarily on temporal representations rather than static aggregates. The highest-ranked components correspond to high-order MFCC means and their standard deviations, which capture subtle variations in harmonic

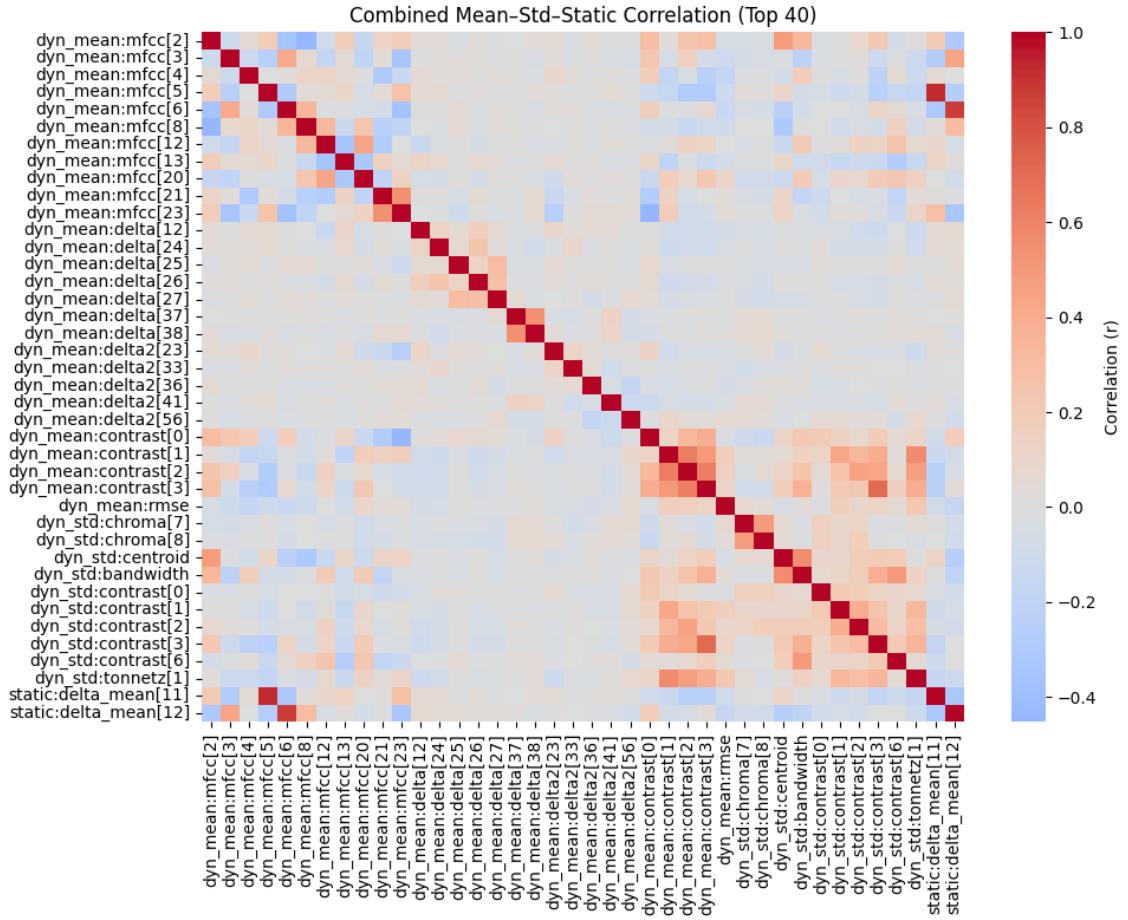


Figure 5.32: Combined Mean–Std–Static Correlation (Top 40)

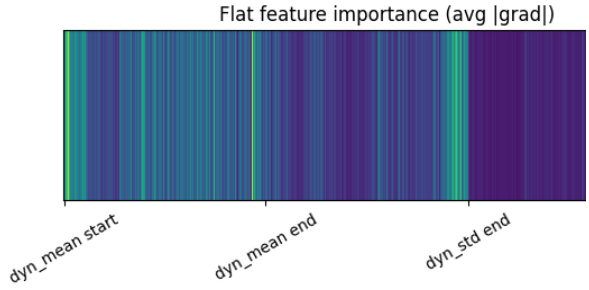


Figure 5.33: Flat feature importance (average |grad| over flattened vector)

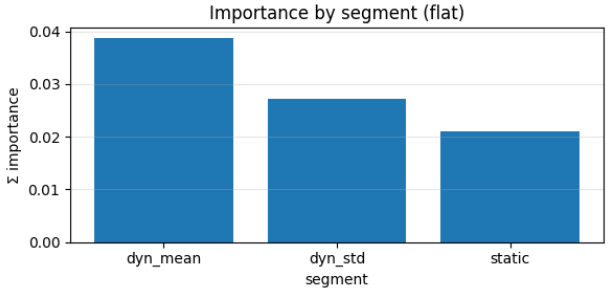


Figure 5.34: Importance by segment (flat)

energy distribution and resonance stability—fundamental indicators of a speaker's vocal tract shape and habitual articulation pattern. In contrast, a few static descriptors such as formant averages and mean pitch statistics appear among the upper ranks but with significantly lower gradient strength, reflecting their supportive rather than primary role. The least influential features include peripheral MFCC coefficients and redundant static measures like global ZCR (Zero-Crossing Rate) or RMSE (Root Mean Square Energy) averages, which add little new information to the discriminative space. This hierarchical weighting shows that the encoder automatically prioritizes physiologically meaningful cues spectral stability and controlled micro-variations while suppressing irrelevant or noisy signals, yielding a robust and

interpretable embedding of speaker identity. For feature importance, in Table 5.30, dynamic mean and standard deviation features dominate the top-ranked importance list, whereas Table 5.31 highlights less influential features.

5.4.6.6 Interpretation and Implications

The hierarchical importance distribution reveals that the encoder's understanding of speaker identity is strongly time-dependent. Dynamic-mean features encode stable spectral envelopes that reflect vocal tract resonance, while dynamic-std features capture fine articulatory micro-variations that distinguish speakers sharing similar pitch or accent. Static features act as global anchors but hold secondary significance once dynamic evidence is established. Together, these complementary cues yield a feature space that aligns closely with established principles of acoustic phonetics stability in formant spacing, energy distribution, and harmonic continuity are all recognized physiological markers of identity. Consequently, the model's verification reliability at extremely low false-acceptance rates is not coincidental but emerges from its alignment with intrinsic properties of human speech production. The network's interpretability, as demonstrated through correlation and gradient attribution, confirms that its decisions are grounded in authentic vocal patterns rather than statistical shortcuts.

Parameter	**Base model**	**Recall-focused model**	**main/Best-overall model**
Training Configuration			
Batch size	128	256	256
Train / Validation split	0.85 / 0.15 (index-level)	0.90 / 0.10	0.85 / 0.15
Epochs	150	3000	2000
Learning rate	Static 0.005	Cosine Annealing 1e-3	Cosine Annealing 1e-3
Scheduler	MultiStepLR ([30, 60, 80], γ=0.3)	Cosine Warm Restarts (T_0=10, T_{mult}=2)	Cosine Warm Restarts (T_0=20, T_{mult}=2)
Focal loss parameters (α, γ)	(1.0, 2.0)	(1.2, 3.0)	(1.5, 2.9)
Training pair ratio (per speaker)	2 positive : 2 negative	7 positive : 14 negative	2 positive : 4 negative
Validation Setup			
Validation setup	Uses random index split (no K-pool)	$K = 6$ clips per speaker $\rightarrow$ 15 positive, 45 negative pairs	$K = 4$ clips per speaker $\rightarrow$ 6 positive, 18 negative pairs
Validation threshold (VAL_THRESHOLD)	0.5 (sigmoid scale)	0.01 (cosine scale [–1, 1])	0.10 (cosine scale [–1, 1])
Testing and Evaluation			
Test pair generation	4 pairs per speaker (2 positive + 2 negative)	Exhaustive + sampled tests (10 pos / 15 neg per speaker)	Exhaustive + sampled tests (10 pos / 15 neg per speaker)
Operating test threshold	0.5	(to be specified)	0.88 (optimized for low FAR)
Learning objective	General baseline	High recall / coverage	Minimize FAR (security-oriented)
Key evaluation regime	Balanced test split	All-vs-all and sampled impostor test	All-vs-all and sampled impostor test
Performance Summary			
Typical outcomes	Acc $\approx$ 0.867, P=0.82, R=0.94 (F1=0.88)	$\uparrow$ Recall (0.9–0.95), slightly lower precision	$\downarrow$ Recall (0.2–0.7), FAR $\approx$ 0–3$\times$10
Behavior summary	Balanced performance baseline	Prioritizes recall and coverage	Prioritizes security (low FAR) over recall

Table 5.29: Model configuration and evaluation summary across three experimental setups. Each model variant is optimized for a different trade-off between recall, precision and false acceptance rate (FAR).

Feature	Segment	Importance
High-order MFCC mean	dyn_mean	0.000527
Spectral Contrast std	dyn_std	0.000493
MFCC mean (mid-frequency)	dyn_mean	0.000489
Bandwidth std	dyn_std	0.000460
MFCC mean (low-frequency)	dyn_mean	0.000386
Delta MFCC mean	dyn_mean	0.000378
Pitch stability mean	dyn_mean	0.000361
Spectral Centroid mean	dyn_mean	0.000358
Spectral Rolloff mean	dyn_mean	0.000351
Formant average (F1–F3)	static	0.000340
HNR std (voice quality)	dyn_std	0.000335
Delta^2 MFCC mean	dyn_mean	0.000326

Table 5.30: Top-ranked features based on average absolute gradient (|grad|), indicating the most influential components in the model.

Feature	Segment	Importance
High-frequency MFCC mean	dyn_mean	0.000009
Formant F3 average	static	0.000010
Spectral Flatness std	dyn_std	0.000010
Pitch range (mean–min)	static	0.000033
ZCR average	static	0.000036
RMSE mean	static	0.000037
HNR average	static	0.000037
VAD probability mean	static	0.000039

Table 5.31: Bottom-ranked features with the lowest average absolute gradient (|grad|), indicating minimal influence on model output.

Chapter 6

Web Deveopment

The speaker verification system is deployed as a comprehensive web application using Flask, a lightweight Python web framework that provides flexibility and simplicity in managing backend processes. This deployment offers an intuitive and interactive interface that allows users to access the system's deep learning models through multiple verification modes. These include file uploads, real-time voice recording and comprehensive audio analysis. The platform integrates five distinct deep learning models that together ensure robust performance: a speaker verification model, an AI voice detection model, an age and gender classification model and finally a regional dialect identification model.

6.1 System Architecture

The architecture of the system follows a client-server model, where the frontend enables seamless user interaction and the backend handles data processing, model inference and feature extraction. The backend of the system is implemented using the Flask framework in Python, while the frontend utilizes a combination of HTML5, CSS3 and JavaScript (ES6+) to ensure responsiveness and user engagement. The user interface is built with Bootstrap 5.3 for a clean and adaptive design. Deep learning functionalities are implemented using the PyTorch framework, while audio preprocessing and transformation rely on the libraries `librosa`, `soundfile` and `pydub`. For model serving, the system performs CPU-based inference to maintain stability and reduce deployment complexity. As shown in Figure 6.1, the system integrates front end and backend components to perform voice-based speaker authentication, AI voice detection and demographic classification.

6.2 Core Components

The web deployment consists of several core components that work collaboratively to ensure accurate speaker verification and smooth system performance. Each component is designed to handle a specific stage of processing, from raw audio acquisition to feature extraction, model inference and result visualization. Together, these components form an integrated framework that supports efficient audio analysis, reliable verification and real-time user interaction within the web application.

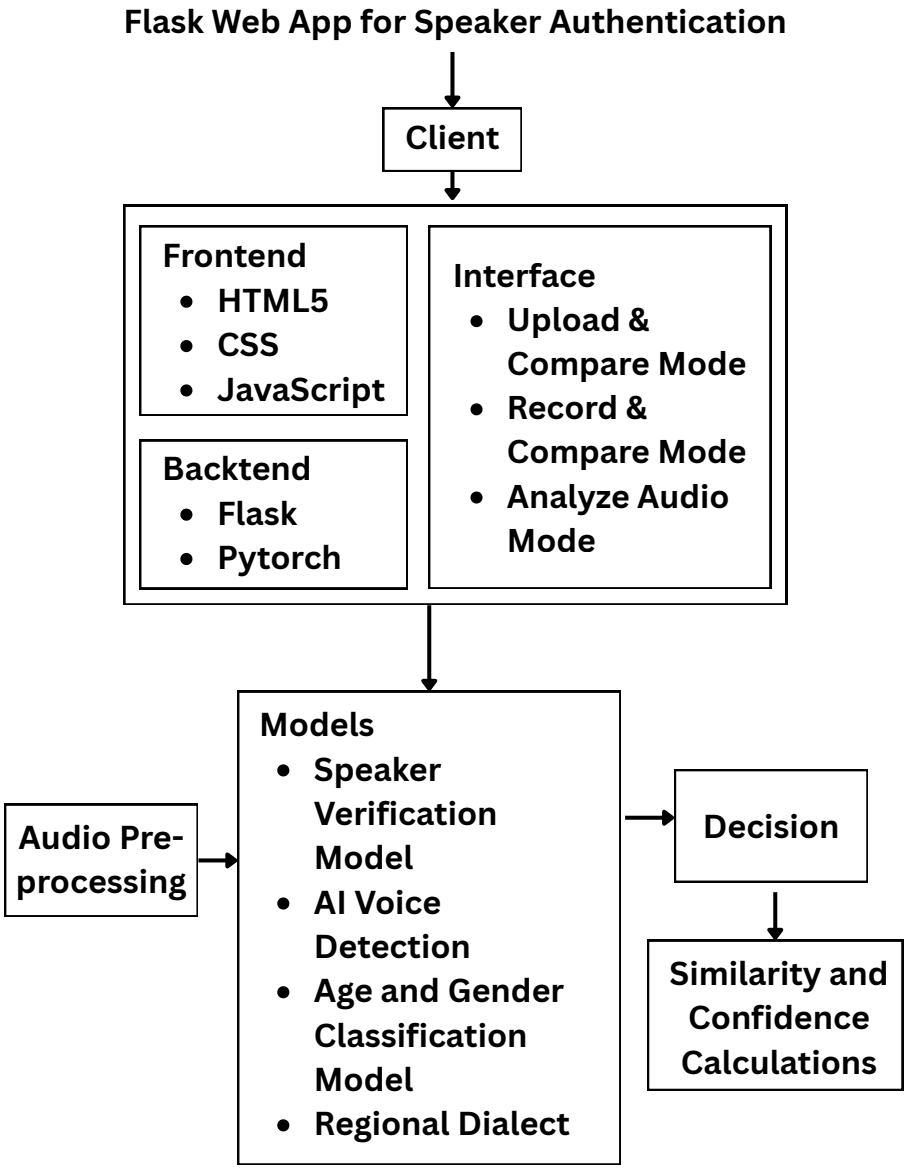


Figure 6.1: Architecture of the proposed web application for voice-based biometric analysis.

6.2.1 Audio Feature Extraction Pipeline

A critical component of the system is the audio feature extraction pipeline, which transforms raw audio signals into high-dimensional numerical feature representations suitable for deep learning models. The pipeline performs a series of preprocessing and feature computation steps designed to capture spectral, prosodic and voice-quality characteristics of a speaker.

During preprocessing, the system extracts a 476-dimensional feature vector from each input audio file. This vector consists of several key feature groups. The Mel-Frequency Cepstral Coefficients (MFCC) contribute 180 dimensions, combining 20 base coefficients with their first and second-order derivatives and including both mean and standard deviation statistics. Spectral features, consisting of seven

parameters such as spectral centroid, rolloff, bandwidth, zero-crossing rate, RMS energy and spectral contrast, represent the energy distribution of the audio signal. Prosodic features add another 18 dimensions, derived from chroma (12) and tonnetz (6) representations, which describe harmonic and tonal properties. Finally, 271 voice-quality features capture characteristics such as pitch variability, formant frequencies, jitter, shimmer and harmonics-to-noise ratio, all of which are crucial for distinguishing individual voices.

6.2.2 Speaker Verification Model

The speaker verification model employs a deep neural network architecture to calculate speaker similarity based on the extracted features. The model is composed of an embedding network followed by a similarity computation network. The embedding network converts feature vectors into fixed-length representations that capture speaker-specific attributes. The similarity computation network then evaluates how closely two embeddings align within a learned feature space. The final similarity score between two embeddings is computed as:

$$S = \sigma\Big(W_4\big(\sigma(W_3(\sigma(W_2(\sigma(W_1F + b_1)) + b_2)) + b_3)\big) + b_4\Big) \tag{6.1}$$

where $F \in R^{512}$ is the concatenated similarity feature vector, W_1, W_2, W_3, W_4 are the learned weight matrices, b_1, b_2, b_3, b_4 are the bias vectors, and σ denotes the sigmoid activation function. The resulting similarity probability $S \in [0, 1]$ indicates the likelihood that two audio samples belong to the same speaker. As shown in Equation (6.1), this formulation allows the model to capture complex relationships between embeddings.

6.2.3 Similarity and Confidence Calculation

To ensure interpretability and reliability, the system computes both similarity and confidence measures. The confidence measure C is derived from the distance of the similarity score S from the decision boundary, formulated as:

$$C = 1 - 2\,|S - 0.5| \tag{6.2}$$

This formulation ensures that the confidence is lowest when $S = 0.5$, representing uncertainty and highest when S approaches 0 or 1. The final verification decision D is determined using a threshold-based rule:

$$D = \begin{cases} 1, & \text{if } S \geq \tau \\ 0, & \text{otherwise} \end{cases}, \quad \tau = 0.5 \tag{6.3}$$

As shown in Equations (6.2) and (6.3), this decision rule guarantees that verification outcomes remain consistent and interpretable.

6.2.4 Multi-Model Integration

The web platform integrates five customized deep learning models to enhance analytical capability and provide comprehensive results. The speaker verification model (proposed Siamese model) accepts 476-dimensional feature vectors and outputs a

similarity probability between 0 and 1, utilizing a multi-layer perceptron architecture with residual connections. The AI voice detection model, based on convolutional neural networks, operates on Mel-spectrogram inputs of size 128×157 and predicts the probability of synthetic or AI-generated speech. The age and gender classification model employs a bidirectional LSTM architecture with an attention mechanism, taking log-Mel spectrograms of size 94×128 as input to predict age group and gender simultaneously. Finally, the regional dialect classification model is based on a Residual network processes raw audio waveforms of 80,000 samples and identifies the dialect across ten possible regional categories.

6.3 Web Interface Implementation

The front end interface of the application offers three principal modes of interaction to accommodate different user needs. In the *Upload and Compare* mode, users can submit pre-recorded audio files in MP3 format. The system processes both files through the feature extraction pipeline and computes a similarity score indicating how closely the two voices match. In the *Record and Compare* mode, users can record their voices directly from the browser using the Web Audio API. The Media Recorder API captures audio in WebM format and converts it into a compatible form for processing through the backend pipeline. The *Analyze Audio* mode provides an advanced diagnostic view by performing AI voice detection, age and gender classification and dialect identification, each accompanied by confidence scores and probability distributions. As illustrated in Figures 6.2–6.4, the web deployment interface presents three primary operational modes: the system interface (Figure 6.2), verification results display (Figure 6.3), and the audio analysis panel (Figure 6.4).

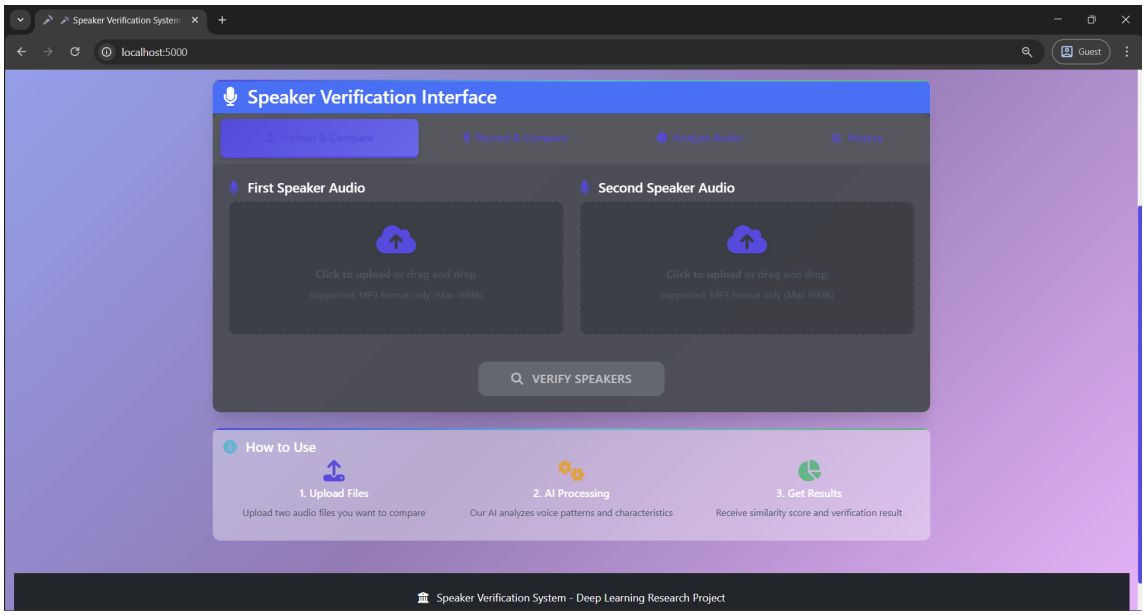


Figure 6.2: System Interface

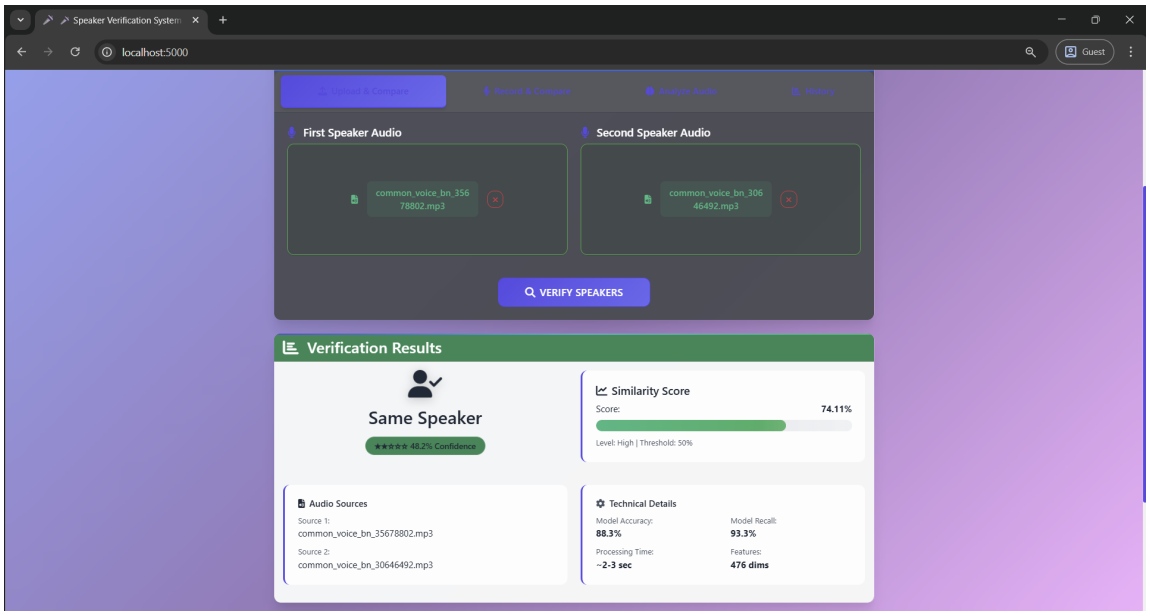


Figure 6.3: Verification Results

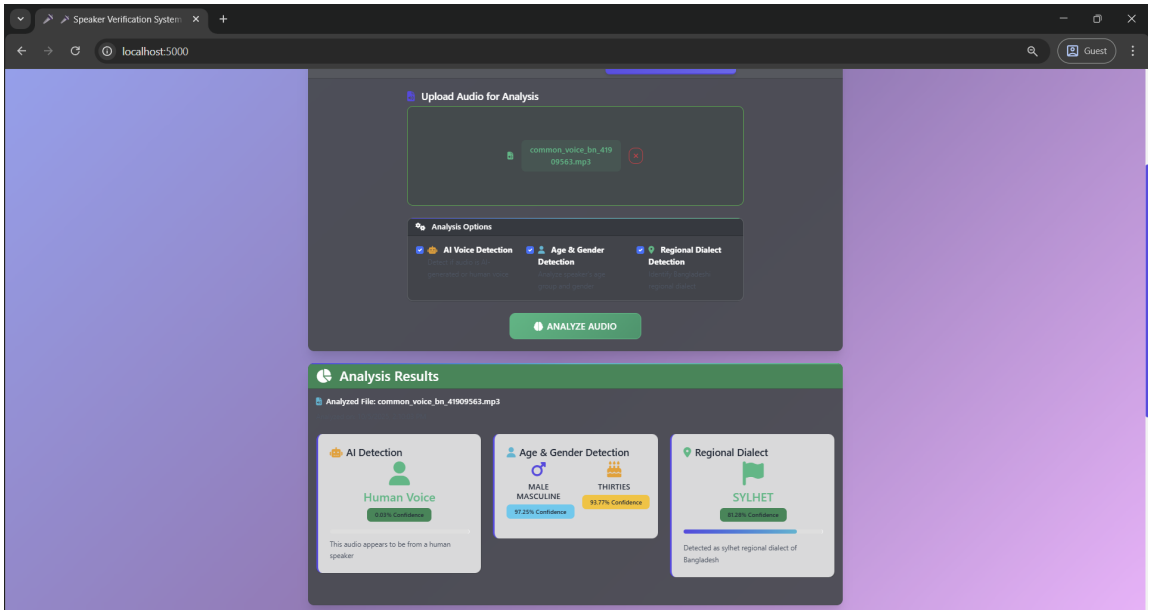


Figure 6.4: Audio Analysis Panel

6.4 Performance Optimization

The deployment implements several optimization techniques to ensure fast response times and efficient use of computational resources. Model loading follows a dynamic strategy, where models are initialized only when required, thus minimizing memory overhead. Audio files are processed using a sequence of optimized fallback mechanisms to handle diverse formats and sampling rates. Initially, audio data are loaded using the `librosa` library, while `soundfile` ensures compatibility with WebM recordings. In cases of unsupported formats, the system employs `pydub` for conversion and normalization.

Memory management is handled through automatic cleanup routines that remove

uploaded files after processing, preventing unnecessary storage accumulation. Tensor operations are optimized for CPU inference using suitable data types, while streaming-based audio processing minimizes the memory footprint.

6.5 Real-Time Processing Pipeline

The real-time processing pipeline follows a structured workflow where audio files are received, validated, processed and passed through the relevant deep learning models. Each stage includes comprehensive error handling mechanisms and fallback procedures to ensure consistent operation across different browsers, devices and audio formats. This modular pipeline design contributes to system robustness and scalability, allowing future extensions such as multilingual support and distributed inference.

6.6 Security and Reliability

Security and reliability are fundamental design priorities for the deployed system. Uploaded files are restricted to a maximum size of 16 MB and only validated audio formats are accepted to prevent malicious file injection. All temporary files are automatically deleted after processing to maintain privacy and storage efficiency. The recording functionality requires an HTTPS connection to ensure secure audio transmission. Furthermore, the system incorporates extensive input validation, exception handling and graceful degradation to maintain operational stability.

6.7 Confidence Visualization

The application provides a set of visual indicators to enhance user comprehension of the results. Similarity scores are presented through progress bars, while confidence levels are represented using a five-star rating mechanism. Probability distributions for multi-class predictions, such as age, gender, and dialect classification, are visualized through bar plots and heatmaps. Additionally, the platform maintains a performance log to track historical outcomes and improve interpretability over time.

The web deployment demonstrates the practical implementation of deep learning models in a real-world speaker verification environment. Through careful integration of Flask-based web services, PyTorch models and advanced audio feature extraction techniques, the system provides accurate, interpretable, and secure verification results. The multi-model integration and optimized processing pipeline ensure both reliability and usability, marking a significant step toward accessible voice-based authentication systems.

Chapter 7

Limitations and Future Work

Although this study successfully develops a Bengali voice signature authentication system using deep learning, several limitations remain that can be addressed in future work.

7.1 Addressing Emotion Detection

The system currently does not include natural language processing (NLP) features to detect or analyze emotions in speech. Incorporating emotion recognition can enable the system to understand tone and mood changes, potentially improving user verification.

7.2 Managing Voice Changes from Health or Stress

The system does not handle voice variations caused by sickness, stress, or fatigue. Future models can be trained to recognize and adapt to these variations based on different physical and environmental factors.

7.3 Improving Cross-Device Performance

The dataset does not account for variations when the same person's voice is recorded using different microphones or devices. Future work can focus on developing models that generalize across devices to ensure consistent performance.

7.4 Enabling Live Web Recording

In the current web version, users can only upload pre-recorded audio clips. A live recording feature can be added to allow direct voice recording for real-time authentication and testing.

7.5 Expanding the Dataset

The system is trained on a small number of recordings. Collecting more voice samples from different ages, genders and regions can improve accuracy.

7.6 Incorporating Regional Dialects

The current dataset lacks voice samples representing all districts and regional dialects across Bangladesh. Future research can focus on collecting and integrating these diverse dialects to enhance the system's inclusiveness and performance for speakers from different regions.

7.7 Scaling for Multiple Users

The system may slow down or have errors when many people use it at the same time. Future work can focus on making it work smoothly for more users.

7.8 Managing the Precision-Recall Tradeoff in the Siamese Model

Future work can explore strategies to balance correct speaker identification and minimizing incorrect matches in the Siamese model. This can include adjusting similarity thresholds, using multiple verification steps, or combining different model outputs to find an optimal tradeoff between precision and recall.

7.9 Handling Noisy or Real-World Conditions

The system may not work well if recordings are noisy or from different devices. Future work can include testing and adjusting the system for real-world conditions.

Addressing these limitations can help build a more adaptable, accurate, and practical Bengali voice authentication system that performs consistently across different users, devices, and environments.

Chapter 8

Conclusion

This thesis presents the development of a Bengali voice signature authentication system for financial applications using deep learning. The study uses two Bengali voice datasets. Different preprocessing methods are applied based on the requirements of each model, such as noise cleaning, data balancing, and metadata validation. Several models are developed and trained for specific tasks: a Modified BiLSTM model for age and gender classification, a CNN model for AI-generated voice detection, a Modified Residual CNN model for identifying regional dialects, and finally a Siamese model for speaker verification.

The system is tested and the results show promising outcomes. The Modified BiLSTM model reaches 98% accuracy for gender and 95% for age classification, the CNN model for AI detection reaches 99% and the Modified Residual CNN for regional dialect classification reaches 94%. For the Siamese model, a tradeoff between precision and recall is observed when the recall rate increases above 90%, the precision drops to around 80% and when the precision approaches 100%, the recall falls to about 3%; even at 95% precision, the recall remains around 25%. These results indicate that while the Siamese model requires further tuning, the overall system performs well across multiple tasks and shows potential for future integration into Bengali financial authentication systems.

The study shows that using more than one deep learning model makes voice-based authentication more accurate and dependable compared to traditional password or PIN systems. Although the system is not yet applied in real financial platforms, it shows strong potential for future use in banking and digital payment systems. By focusing on the Bengali language, the research helps reduce the lack of local-language datasets and promotes the inclusion of regional languages in AI-based technologies. The framework developed in this study can be expanded further by adding transformer-based or self-learning models to improve performance in noisy and diverse environments. Overall, this research represents an important step toward creating a secure and language-inclusive authentication method for Bengali speakers. It demonstrates that deep learning can be effectively applied to real-world identity verification while respecting linguistic diversity. With continued development and integration, this system can support the future of digital banking, e-governance, and customer verification in Bangladesh, helping to build a safer and more trusted digital society.

Bibliography

[1] E. S. Mohamed, H. I. Abd Elkader, and A. A. Abd El-Latif. "Speaker identification for OFDM-based aeronautical communication system". In: *International Journal of Speech Technology* 22.1 (2019), pp. 51–62. DOI: 10.1007/s10772-018-9538-9. URL: https://doi.org/10.1007/s10772-018-9538-9.

[2] The Daily Star. "BFIU Annual Report: Suspicious transaction, activity reports up 65pc". In: *The Daily Star* (2024). URL: https://www.thedailystar.net/business/news/suspicious-financial-activities-rise-65pc-2022-23-3424371.

[3] A. H. Khan and P. S. Aithal. "Voice biometric systems for user identification and authentication – A literature review". In: *International Journal of Applied Engineering and Management Letters (IJAEML)* 6.1 (2022), pp. 198–209. DOI: https://doi.org/10.5281/zenodo.6471040. URL: https://doi.org/10.5281/zenodo.6471040.

[4] N. Markl and C. Lai. "Context-sensitive evaluation of automatic speech recognition: Considering user experience & language variation". In: *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing* (2021), pp. 34–40.

[5] Y. Xie et al. "Large-scale multichannel transformer-based speaker identification with knowledge transfer using harmonic-percussive source separation". In: *2022 31st Wireless and Optical Communications Conference (WOCC)* (2022), pp. 1–5. DOI: https://doi.org/10.1109/WOCC55104.2022.9880598. URL: https://doi.org/10.1109/WOCC55104.2022.9880598.

[6] M. K. Singh et al. "Speaker Recognition Assessment in a Continuous System for Speaker Identification". In: *IJEER* 10.4 (2022), pp. 862–867. DOI: https://doi.org/10.37391/IJEER.100418. URL: https://doi.org/10.37391/IJEER.100418.

[7] A. Mehrish et al. "A review of deep learning techniques for speech processing". In: *arXiv* (2023). URL: https://ar5iv.labs.arxiv.org/html/2305.00359.

[8] Y. Yang et al. "Speaker verification in agent-generated conversations". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics* 1 (2024), pp. 5655–5676.

[9] Z. Chen et al. "Large-scale self-supervised speech representation learning for automatic speaker verification". In: *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2022), pp. 6147–6151.

[10] M. Biswas et al. “Automatic spoken language identification using MFCC based time series features”. In: *Multimedia Tools and Applications* 82 (2023), pp. 9565–9595. DOI: https://doi.org/10.1007/s11042-021-11439-1. URL: https://doi.org/10.1007/s11042-021-11439-1.

[11] K. A. R. Arnab et al. “Shohojogi: An automated Bengali voice chat system for banking customer services”. In: *Brac University* (2023).

[12] A. H. Khan and P. S. Aithal. “ABCD analysis of voice biometric system in banking”. In: *International Journal of Management, Technology, and Social Sciences (IJMTS)* 9.2 (2024), pp. 1–17. DOI: https://doi.org/10.5281/zenodo.10963565. URL: https://doi.org/10.5281/zenodo.10963565.

[13] Jinyu Li et al. “An Overview of Noise-Robust Automatic Speech Recognition”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.4 (2014), pp. 745–777. DOI: 10.1109/TASLP.2014.2304637.

[14] Massimiliano Todisco et al. “ASVspoof 2019: Future Horizons in Spoofed and Fake Audio Detection”. In: *Proceedings of Interspeech 2019*. 2019, pp. 1008–1012. DOI: 10.21437/Interspeech.2019-2249.

[15] Christian Anthony Julius, S Vijayalakshmi, and Tiny S Palathara. “Spoken Language Identification using Deep Learning”. In: *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)*. Vol. 1. 2024, pp. 1–7. DOI: 10.1109/ICEECT61758.2024.10738935.

[16] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. “VoxCeleb2: Deep Speaker Recognition”. In: *arXiv preprint arXiv:1806.05622* (2018). DOI: 10.48550/arXiv.1806.05622.

[17] Nicholas Evans, Junichi Yamagishi, and Tomi Kinnunen. “Spoofing and Countermeasures for Speaker Verification: A Need for Standard Corpora, Protocols and Metrics”. In: *IEEE Signal Processing Society Newsletter* (May 2013).

[18] J. Yu, A. de Antonio, and E. Villalba-Mora. “Deep learning (CNN, RNN) applications for smart homes: A systematic review”. In: *Computers* 11.2 (2022), p. 26. DOI: https://doi.org/10.3390/computers11020026. URL: https://doi.org/10.3390/computers11020026.

[19] Gnani.ai. *Voice Biometrics in Banking: Securing Customer Interactions*. https://www.gnani.ai/resources/blogs/voice-biometrics-in-banking-securing-customer-interactions-bb31d. Accessed: 2025-10-07. 2025.

[20] Telnyx. *AI Voice Biometrics for Fraud Detection and Security*. https://telnyx.com/resources/ai-voice-biometrics. Accessed: 2025-10-07. 2025.

[21] NICE. *Voice Biometrics for Contact Centers*. https://www.nice.com/info/voice-biometrics-for-contact-centers. Accessed: 2025-10-07. 2025.

[22] ClearML. *What is Deep Learning?* https://clear.ml/blog/what-is-deep-learning. Accessed: 2025-10-07. 2025.

[23] GeeksforGeeks. *Applications of Deep Learning*. https://www.geeksforgeeks.org/deep-learning/deep-learning-examples/. Accessed: 2025-10-07. 2025.

[24] Y. Chen. "Deep Learning in Financial Fraud Detection: Innovations and Applications". In: *Journal of Financial Technology* 4.1 (2025). Accessed: 2025-10-07, pp. 1–15. DOI: 10.1016/j.jfintec.2025.1000372. URL: https://www.sciencedirect.com/science/article/pii/S2666764925000372.

[25] Amazon Web Services, Inc. *What is Deep Learning?* https://aws.amazon.com/what-is/deep-learning/. Accessed: 2025-10-07. 2025.

[26] Xiwei Huang. "Study for Automatic Speech Recognition for Wav2Vec2.0". In: *Applied and Computational Engineering* 158 (2025). Accessed: 2025-10-07, pp. 96–102. DOI: 10.54254/2755-2721/2025.TJ23214. URL: https://www.ewadirect.com/proceedings/ace/article/view/23214.

[27] Zejiang Hou et al. "Revisiting Convolution-Free Transformer for Speech Recognition". In: *Interspeech 2024*. Accessed: 2025-10-07. 2024. URL: https://www.isca-archive.org/interspeech_2024/hou24_interspeech.pdf.

[28] Multiple Authors. "Speaker-Attributed Training for Multi-Speaker Speech Recognition Using Multi-Stage Encoders and Attention-Weighted Speaker Embedding". In: *Applied Sciences* 14.18 (2024). Accessed: 2025-10-07, p. 8138. DOI: 10.3390/app14188138. URL: https://www.mdpi.com/2076-3417/14/18/8138.

[29] Danwei Cai and Ming Li. "Leveraging ASR Pretrained Conformers for Speaker Verification through Transfer Learning and Knowledge Distillation". In: *IEEE Transactions on Audio, Speech, and Language Processing* (2024). Accessed: 2025-10-07. DOI: 10.1109/TASLP.2024.3419426.

[30] Harunori Kawano and Sota Shimizu. "An Effective Transformer-Based Contextual Model and Temporal Gate Pooling for Speaker Identification". In: *arXiv preprint* (2023). Accessed: 2025-10-07. URL: https://arxiv.org/abs/2308.11241.

[31] K. J. Devi and K. Thongam. "Automatic speaker recognition from speech signal using bidirectional long-short-term memory recurrent neural network". In: *Computational Intelligence* 39.10 (2020).

[32] M. Fasounaki et al. "A Comparative Assessment of Text-independent Automatic Speaker Identification Methods Using Limited Data". In: *European Journal of Science and Technology* 26 (2021), pp. 217–222.

[33] A. Rizvi et al. "Cross-lingual speaker identification for Indian languages". In: *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing* (2023), pp. 979–987. URL: https://aclanthology.org/2023.ranlp-1.124.

[34] P.-H. Chi et al. "Audio ALBERT: A lite BERT for self-supervised learning of audio representation". In: *Papers with Code* (2020). URL: https://paperswithcode.com/paper/audio-albert-a-lite-bert-for-self-supervised.

[35] J. Zhu, M. Hasegawa-Johnson, and L. Sari. "Identify speakers in cocktail parties with end-to-end attention". In: *Papers with Code* (2020). URL: https://paperswithcode.com/paper/identify-speakers-in-cocktail-parties-with-end.

[36] V. Brydinskyi et al. "Comparison of modern deep learning models for speaker verification". In: *Applied Sciences* 14.4 (2024), p. 1329. DOI: https://doi.org/10.3390/app14041329. URL: https://doi.org/10.3390/app14041329.

[37] N. Hervé et al. “Using ASR-generated text for spoken language modeling”. In: *Proceedings of BigScience Episode 5 — Workshop on Challenges & Perspectives in Creating Large Language Models* (2022), pp. 17–25. DOI: https://aclanthology.org/2022.bigscience-1.2/. URL: https://aclanthology.org/2022.bigscience-1.2/.

[38] D. Xu et al. “Introducing semantics into speech encoders”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics* 1.Long Papers (2023), pp. 11413–11429. DOI: https://aclanthology.org/2023.acl-long.639.pdf. URL: https://aclanthology.org/2023.acl-long.639.pdf.

[39] T. Yu et al. “Speech-text dialog pre-training for spoken dialog understanding with explicit cross-modal alignment”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics* 1 (2023), pp. 7900–7913. DOI: https://aclanthology.org/2023.acl-long.438.pdf. URL: https://aclanthology.org/2023.acl-long.438.pdf.

[40] A. Caubrière et al. “Where are we in named entity recognition from speech?” In: *Proceedings of the Twelfth Language Resources and Evaluation Conference* (2020), pp. 4514–4520. DOI: https://aclanthology.org/2020.lrec-1.556. URL: https://aclanthology.org/2020.lrec-1.556.

[41] N. Markl and C. Lai. “Context-sensitive evaluation of automatic speech recognition: Considering user experience & language variation”. In: *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing* (2021), pp. 34–40. DOI: https://aclanthology.org/2021.hcinlp-1.6. URL: https://aclanthology.org/2021.hcinlp-1.6.

[42] M. Bartolo. “Deep learning for speaker identification: Architectural insights from AB-1 corpus analysis and performance evaluation”. In: *Papers with Code* (2024).

[43] Mozilla Foundation. *Mozilla Common Voice Corpus 20.0*. A publicly available multilingual voice dataset containing short speech clips and transcriptions, including Bengali language samples. Mozilla Foundation, Nov. 2024. URL: https://commonvoice.allizom.org/cv/datasets.

[44] NVIDIA Corporation. *NVIDIA Riva: Speech AI SDK for Real-Time Automatic Speech Recognition and Text-to-Speech*. https://developer.nvidia.com/riva. Accessed: 2025-10-09. 2024.

[45] Bengali.AI. *ASR for Regional Dialects – Ben10 Competition Dataset*. https://www.kaggle.com/competitions/ben10/data. Accessed: 2025-06-18. 2024.