A Two-Stage Framework for Dialectal Bengali ASR and Standardization using Whisper and BanglaT5

By

SHOWROV AZAM

221-15-5843

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by

Amit Chakraborty Chhoton

Assistant Professor

Department of Computer Science and Engineering

Daffodil International University

Co-Supervised by

Ms. Sharun Akter Khushbu

Assistant Professor

Department of Computer Science and Engineering

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

Dhaka, Bangladesh

January 06, 2026

APPROVAL

This Project titled **"A Two-Stage Framework for Dialectal Bengali ASR and Standardization using Whisper and BanglaT5"**, submitted by Name, ID No: **221-15-5843** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **6 January 2026**.

BOARD OF EXAMINERS

Professor Dr. Md. Fokhray Hossain (MFH) — **Chairman**
Dean & Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Ms. Samia Nawshin (SN) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Israt Jahan (IJN) — **Internal Examiner**
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Nazibur Rahman — **External Examiner**
Technical Lead - Database Administrator
Telenor - Grameen Phone Account

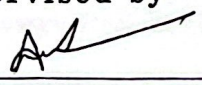
©Daffodil International University

Scanned with CamScanner

DECLARATION

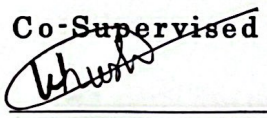
We hereby declare that this project has been done by us under the supervision of **Amit Chakraborty Chhoton, Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Amit Chakraborty Chhoton
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:



Ms. Sharun Akter Khushbu
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Showrov Azam

Showrov Azam
Student ID: 221-15-5843
Department of Computer Science and Engineering
Daffodil International University

©Daffodil International University

Scanned with CamScanner

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Amit Chakraborty Chhoton**, **Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of Natural Language Processing (NLP) to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

Although machines now understand regular Bengali almost like people do, people using local dialects still get left behind. Because these ways of speaking shift sounds, use different words, or change tone, mainstream voice tools fail badly - often completely breaking down. On top of that, past work zoomed in on one-off transcriptions or just one dialect at a time, skipping the challenge of making such speech clear to outsiders. We built a clear two-part approach to link speech variety identification with meaning translation across ten Bengali forms - Sylhet, Chittagong, Rangpur, Sandwip, Narail, Narsingdi, Barishal, Tangail, Kishoreganj, and Habiganj - to tackle this issue. In addition, we gathered and reviewed more than 13,000 voice samples. Such data is uncommon since labeled instances are typically missing. Our strategy applies an adjusted OpenAI Whisper Medium system to transcribe spoken words. Best result showed a Word Error Rate (WER) of 18.47%. After that, a translation system based on BanglaT5 changes the converted speech into proper Bengali. What sets this study apart is a new "Smart Fix" mix-method tool made to fix grammar issues tied to context - like wrong use of informal or formal pronouns (Tui vs Tumi), along with mixed-language mistakes - that deep learning systems often struggle with when data is limited. While the translator scored a BLEU of 5.11 and showed a validation loss at 2.02, the project builds a full reference point for handling multiple regional accents in voice recognition. The results demonstrate that combining steps in stages, using both rule-based and learned methods, gives a realistic way forward to include more people across Bangladesh who speak different versions of the language.

Table of Contents

Approval i

Declaration ii

Acknowledgements iii

Abstract iv

List of Figures vii

List of Tables x

1 Introduction 1

1.1 Introduction 1

1.2 Motivation 1

1.3 Objectives 2

1.4 Methodology 2

1.5 Project Outcome 2

1.6 Organization of the Report 2

2 Background 4

2.1 Introduction 4

2.2 Literature Review 4

2.3 Gap Analysis 9

2.4 Summary 10

3 Research Methodology 11

3.1 Methodology/Requirement Analysis & Design Specification 11

3.1.1 Overview 11

3.1.2 Data Collection and Description 11

3.1.3 Data Preprocessing 11

3.1.4 Exploratory Data Analysis 12

3.1.5 Train-Test Duration analysis 34

3.1.6 Proposed Methodology 34

3.1.7 Functional and Nonfunctional Requirements 39

3.1.8 Data Flow Diagram 40
3.1.9 Ui Design 42
3.2 Detailed Methodology and Design 42
3.3 Project Plan 43
3.4 Task Allocation 43
3.5 Summary 44

4 Implementation and Results 45
4.1 Environment Setup 45
4.2 Testing and Evaluation/Performance/ Comparative Analysis 45
4.3 Results and Discussion 46
4.4 Summary 48

5 Engineering Standards and Design Challenges 49
5.1 Compliance with the Standards 49
5.1.1 Software Standards 49
5.1.2 Hardware Standards 44
5.1.3 Communication Standards 50
5.2 Impact on Society, Environment and Sustainability 50
5.2.1 Impact on Life 50
5.2.2 Impact on Society & Environment 51
5.2.3 Ethical Aspects 51
5.2.4 Sustainability Plan 51
5.3 Project Management and Financial Analysis 51
5.4 Complex Engineering Problem 52
5.4.1 Complex Problem Solving 53
5.4.2 Engineering Activities 54
5.5 Summary 56

6 Conclusion 57
6.1 Summary 57
6.2 Limitation 57
6.3 Future Work 58

References 59

List of Figures

Figure 1: Quantitative distribution of audio samples across ten districts.....13

Figure 2: Comparative analysis of Sample Size versus Lexical Diversity.....14

Figure 3: Lexical composition of the dialectal corpus. (Left) Absolute vocabulary size (unique word types) per district. (Right) Distribution of the top 20 most frequent tokens across the entire dataset.....15

Figure 4: Word cloud visualization of the corpus vocabulary.....16

Figure 5: Temporal distribution analysis of the audio corpus. (Left) Global histogram of audio segment lengths with a kernel density estimate. (Right) Box-plot comparison of duration variance across dialectal regions.....17

Figure 6: Acoustic feature analysis of the dialectal corpus. (Left) Mean percentage of silence per audio file with standard deviation bars. (Right) Violin plots represent the probability density of the Fundamental Frequency across districts.....18

Figure 7: Distribution of Signal-to-Noise Ratio (SNR) across the dataset (N=5000 samples).....19

Figure 8: Analysis of temporal-linguistic correlation.....20

Figure 9: Time-domain waveform and Mel-spectrogram of the Barishal dialect, exhibiting distinct silence intervals and robust lower-formant structures.....20

Figure10:Listen to how Chittagong sounds - waves stay close together,energy holds firm across tones, which suggests people talk quick here.....21

Figure11:Wave shape plus sound map from Habiganj speech - smooth volume shifts, dist inct harmonic levels show up well.....21

Figure 12: Spectral view of the Kishoreganj dialect, showing clear segmented speech parts and strong base-frequency peaks.....22

Figure 13: Time-frequency representation of the Narail dialect, showing clear separation between speech onsets and the noise floor with distinct formant tracking.....22

Figure 14: Acoustic features of the Narsingdi dialect, revealing significant amplitude variance and intense spectral density across the bandwidth.....23

Figure 15: Waveform and Mel-spectrogram of the Rangpur dialect, characterized by impulsive amplitude spikes and distinct vowel formant trajectories.....23

Figure 16: Spectral analysis of the Sandwip dialect, depicting a continuous amplitude envelope and consistent high-frequency energy distribution.....24

Figure 17: Acoustic profile of the Sylhet dialect, with strong waveform peaks and deep spectral lines that reflect its complex intonation.....24

Figure 18: Time-domain and spectral visualization of the Tangail dialect, illustrating a varied prosodic contour with stable harmonic structures in voiced regions.....25

Figure 19: Spectro-temporal analysis of the Barishal dialect. The white spectral centroid trajectory (C_t) demonstrates stability during voiced segments, with energy concentrated in the lower frequency bands (<1024 Hz).....26

Figure 20: Log Power Spectrogram of the Chittagong dialect.....27

Figure 21: Spectral analysis of the Habiganj dialect.....28

Figure 22: Time-frequency representation of the Kishoreganj dialect. The centroid trajectory accurately delineates the dynamic range between silence intervals and high-intensity plosive onsets.....29

Figure 23: Spectrogram of the Narail dialect with overlaid centroid.....30

Figure 24: Acoustic analysis of the Narsingdi dialect.....31

Figure 25: Spectral features of the Rangpur dialect.....31

Figure 26: Log Power Spectrogram of the Sandwip dialect.....32

Figure 27: Spectro-temporal analysis of the Sylhet dialect.....33

Figure 28: Spectral analysis of the Tangail dialect.....33

Figure 29: Comparative analysis of temporal distributions between Training and Testing partitions.....34

Figure 30: Schematic workflow of the proposed two-stage dialect conversion system...35

Figure 31: Data Flow Diagram of the System..... 41

Figure 32: Graphical Interface of the UI..... 42

Figure 33: Training dynamics of the Whisper Medium model.....46

Figure 34: (Left) Validation Loss decays over 10 epochs. (Right) Correlation between WER reduction and BLEU score improvement.....47

List of Tables

Table 1: Summary of Literature Reviewed.....6

Table 2: Gap Analysis Table 9

Table 3: Training Configuration of Whisper and BanglaT5 38

Table 4: Task Allocation Table 44

Table 5: Cost Estimation Table 52

Table 6: Mapping with Complex Engineering Problem..... 53

Table 7: Mapping with knowledge Profile. 53

Table 8: Mapping with Complex Engineering Activities. 54

Chapter 1

Introduction

This section sets up the background, key issues, plus the overall plan for the study. It explains the gap in voice tech across dialects, why closing that gap matters, also what goals and methods were used to build a strong system for recognizing multiple dialects and making them consistent.

1.1 Introduction

There are many ways to say things in Bengali not just the common method of the streets that most people are familiar with. About 260 million people use this language in the world. There are numerous local forms in Bangladesh - Sylheti, Chittagonian, Rangpuria, Barishaila, Swandipian etc. These forms are not only different in accent but also differ in words and rhythm. Most of the modern speech technologies work well with the standard Bengali language, but have difficulty with these local styles. Because of this, there are many difficulties using voice based systems in schools, hospitals, and government offices. This piece of work is an effort to remove that gap by turning daily spoken language into the text version and then converting it into simple standard Bengali audio so that more people could easily access digital and online services.

1.2 Motivation

- This thesis basically focuses on regional dialectal audio recognition.
- It is built in a way that it can support both regional and standard text.
- Collecting and normalizing the local audio data from different places in Bangladesh are difficult.
- The standard models often miss the regional dialects.
- Even OpenAI and Google can not understand Bangla local language sometimes.

1.3 Objectives

The principal objectives of this project are given below:

- To build an ASR system across ten district dialects. Generating dialectal text from dialectal audios.
- Converting Dialectical Bangla Transcript to Standard Bangla Transcript using Hyper-Parameter tuned BanglaT5.
- To build a post processing tool. We named it Smart Fix.

1.4 Methodology

We explored a novel and also a unique technique for adapting Transformer based models to common local and dialect aware accent in everyday speech. It utilizes OpenAI Whisper Medium to catch the spoken words and convert that speech and rural Bangladeshi speech into readable text without any further training, instead of a heavy setup. Next, the model for BanglaT5 is fine-tuned to convert a rural or informal phrase to standard Bengali transcript. Once the accuracy was no longer increasing, a simple rule-based system - Smart Fix was applied in my thesis. The system searches for spurious words and corrects clumsy phrases using some simple rules. It is tested on 13,000 samples that are not automatically selected. The performance is measured in terms of WER and BLEU scores so the results may be compared fairly with other systems.

1.5 Project Outcome

The completed research brings several clear results:

- Improved Accessibility (Direct Dialectal Speech to Standard Text).
- A reliable Dialectal ASR pipeline (Whisper Medium and BanglaT5).
- Novel Algorithmic Contribution (Post Process-Smart Fix).

1.6 Organization of the Report

The rest of the report flowing:

Chapter 2: Background In this section, we discussed background study.
Chapter 3: Research Methodology Here the full research methodology is explained
Chapter 4: Implementation and Results In this part, the Results and Metrices shown.

Chapter 5: Engineering Standards and Design Challenges In this section there are some important discussions about the Standards and Design Challenges of my thesis work.

Chapter 6: Conclusions It is the endpoint of the entire thesis work and any type of future works are written here.

Chapter 2

Background

2.1 Introduction

This chapter aims at presenting an overview of research scenario of Automated Speech Recognition for low resource languages (such as Bangla) and in particular, acoustic and linguistic characteristics of dialects under Bengali along with the methods towards semantic normalization. It shows us how the technology shifted from basic stats to full deep learning setups, but yet, brings up unequal access to people who speak their local dialects. It also points out the important research gaps; no combined pipeline or framework of a robust dialect transcription and semantic correction, and our proposed cascaded system (Whisper Medium followed by BanglaT5) as one of the most promising and nice scoring approaches to cover this research gap.

2.2 Literature Review

Bengali voice to text tech changing fast - from clunky and rule-heavy set ups - to smooth data-powered neural nets. Early gadgets like CMU Sphinx or Microsoft SAPI used HMM-GMM tricks to get isolated words [1], [2]. Those first tries had shown that machines could handle the sounds of Bengali [3], but leaned way too hard on hand-built sound lists - and choked on the sounds of virtuous discourse [4]. Because of that mess, deep learning emerged to paternal allow core flaws rather quickly, after which RNN and then LSTMs cranked up, and brought back results on loopy speech sets such as SUBAK.KO[5]. Now, transformer models are evolving daily. Wav2Vec-BERT with Whisper of OpenAI demonstrate impressive results learning from limited data - likely due to training on different languages [6]. Their quieter counterparts achieved almost 9.05% word error rates in difficult situations such as medical talk, by taking down rare words [7].

The state of the art has improved when it comes to regular Bengali speech recognition but, as is the case with any other language, machines struggle to cope with local dialects - mostly because good data is scarce and pronunciation varies a lot differently (think how "man" sounds near different folks). The real challenge? So, the real challenge is, picking out bits of sound which are the core so we can group them inside the main set. Accents such as Sylheti or Chittagonian [8] stumbling over normal tools and spitting up errors of 70-80%. That hole widens given that the majority voice training leans from simply one way of talking and puts fresh efforts into gathering

long talk samples. Take RegSpeech12 [9], taking an natural chat from 12 zones; while BRADS [10] maping the use of words at eight spots. In order to deal with the fact of these dialects sounds so different, they attempted to combine techniques enhanced by tools like WavLM - ones that pick up on patterns directly from unprocessed audio and often work better when noise or recordings of less-than-perfect quality messes with the clarity of speech [11], [12]. Still, there are, some models, even configurations such as BanglaTalk, a system that converts text into ten regional accents in real time [8], only match words without understanding deeper sense, which makes it very difficult for people who don't understand standard Bengali to really understand what's being said.

Today, to convert local vernacular to regular Bengali language, the experts are relying on the latest smart translation mechanisms with web-based clearing of the text. Instead of old school iron clad rules - like matching wordlists and grammar tools for dialects like Chittagonian language, they're making a shift [13]. Still, grasping words with many possible meanings based on situation is difficult. Lately, the newer models using attention-driven Seq2Seq Transformers have taken a step up as stronger models. BanglaT5 crushed earlier results on structured text tasks - conquering broad models like mBART50 and mT5 with the biggest open dataset of Bengali - approximately 27.5 GB [14]. Lately, on some setups, there was an attempt to reduce disparities between regional modes of speech and the predominant form; but on others, the motivations were inverted so that formal speech was adjusted to accommodate local tones, simply in order to maintain diversity [15]. Large sets such as ONUBAD contributed to the generalization of training for translation facilities, by providing matched texts with one next to the other [16]. For the purposes of voice creation, investigators considered using flow and beat in dialects such as Mymensinghiya - but those studies leaned more to collecting audio samples rather than setting down specific rules for collecting accents [17],[18].

A look at the previous research reveals some obvious gaps - to date, there's been nothing that could manage several Bangla dialects while separating comprehension of speech, meaning and structure all together. The BanglaDialecto effort constructed an entire chain using Whisper and large language models; even there it prioritized just the Noakhali speech but did not address other speech [19]. In contrast, other tools, for example, BanglaTalk, take care of local accents; however, they cannot convert them into standard written form [8]. Another issue: modern neural setups often tend to struggle in the presence of rare languages when a lack of tanness data is coupled with a resulting ability of neural systems to give fake outputs or make slips depending on the context - mixing up the casual forms of you (তুই vs. তুমি) for example. [20] Here, we try to solve this problem by taking the direct approach by using a two-step process: first, a tuned Whisper (Medium) converts speech to text and then, a modified BanglaT5 lines up meaning on 10 regional types. What makes this different? Beyond that it toss out of the way right away a feature named 'Smart Fix' which is

aimed at wiping out grammar hiccups as well as mix-ups from spelling or word picks brought by switching codes - things that trip up even high-end auto systems, both stages working together to push forward actual progress bridging between how people speak locally versus standard Bengali.

The following Table 1 highlights the full literature review of this thesis.

Table 1: Summary of Literature Reviewed.

Author (s)	Year	Title	Methodology	Key Findings
Nasib et al [1].	2018	A Real Time Speech to Text Conversion Technique for Bengali Language.	Statistical ASR using CMU Sphinx 4 (HMM/GMM) with a phonetic dictionary.	Achieved an average accuracy of 71.7% on a small dataset of 10 speakers but struggled with real-world noise and continuous speech.
Sultana et al [2].	2012	Bangla Speech-to-Text Conversion using SAPI.	Rule-based recognition using Microsoft SAPI 5.1 with limited vocabulary.	Demonstrated feasibility for isolated command recognition but failed to handle continuous conversational speech effectively due to SAPI's limitations.
Sayem [3].	2014	Speech analysis for alphabets in Bangla language: automatic speech recognition.	Feature extraction using MFCC combined with Dynamic Time Warping (DTW) and K-Nearest Neighbors (KNN).	Achieved 90% accuracy for isolated spoken letter recognition, proving that hybrid statistical methods can improve robustness over simple template matching.
Mridha et al [4].	2022	A study on the challenges and opportunities of speech recognition for Bengali language.	Comprehensive survey of existing ASR methodologies (HMM, GMM, DNN).	Identified lack of diverse corpora and dialectal variation as the primary bottlenecks for Bengali ASR progress.
Noor et al [20].	2020	A Real Time Bangla Local Language Recognition from Voice.	Google Speech API integration for real-time recognition of local words.	Successfully recognized specific local words but lacked a mechanism to translate or standardize them into formal Bengali.
Kibria et al [5].	2022	Bangladeshi Bangla speech corpus for automatic speech recognition research.	Development of "SUBAK.KO," a large-scale speech corpus trained with RNN-CTC.	Established a baseline for continuous speech recognition in Bangladeshi Bangla, highlighting the need for larger datasets.

Ridoy et al [6].	2025	Adaptability of ASR Models on Low-Resource Language: A Comparative Study of Whisper and Wav2Vec-BERT on Bangla.	Comparative analysis of fine-tuned Whisper (Small/Large) vs. Wav2Vec-BERT on Common Voice & OpenSLR datasets.	Wav2Vec-BERT achieved lower WER (14.42%) than Whisper Large-V2 (28.86%) on standard Bengali, but Whisper showed better zero-shot adaptability.
Kabir et al [7].	2024	Automatic Speech Recognition for Biomedical Data in Bengali Language.	Fine-tuning Whisper (Small) on a custom 46-hour biomedical corpus including Sylheti dialect.	Achieved a WER of 9.05% on medical domain data, proving Whisper's efficacy for specialized, low-resource vocabularies.
Hasan & Dipta [8].	2025	BanglaTalk: Towards Real-Time Speech Assistance for Bengali Regional Dialects.	Client-server architecture using fine-tuned IndicWav2Vec for 10 regional dialects.	Achieved a WER of 74.1% on the RegSpeech12 dataset, enabling real-time dialect recognition but without semantic standardization.
Kumar et al [11].	2025	Jointly Improving Dialect Identification and ASR in Indian Languages using Multimodal Feature Fusion.	Multimodal framework combining Bottleneck Encoder and RoBERTa for joint ASR and Dialect Identification (DID).	Improved DID accuracy to 81.63% and ASR performance (WER 17.73%) across 8 Indian languages by fusing acoustic and textual features.
Hassan et al [9].	2025	RegSpeech12: A Regional Corpus of Bengali Spontaneous Speech Across Dialects.	Creation of a spontaneous speech corpus covering 12 distinct regional dialects of Bangladesh.	Provided a critical resource for dialectal research, revealing significant performance drops in standard ASR models when tested on this data.
Aiman et al [10].	2025	BRADS and BRWDS: Multipurpose audio and text datasets for automatic Bangla regional speech recognition.	Collection of 298 regional words from 8 administrative divisions to capture lexical diversity.	Documented distinct vocabulary variations (e.g., "Mui" vs. "Ami") across regions, supporting the need for dialect-aware models.
Biswas et al [12].	2025	A Unified Denoising and Adaptation Framework for Self-Supervised Bengali Dialectal ASR.	Self-supervised learning using WavLM with a masked speech denoising objective.	Demonstrated that pre-training on noisy dialectal data significantly improves robustness compared to standard Wav2Vec 2.0.
Milon et al [13].	2020	A Comprehensive Dialect Conversion Approach from Chittagonian to Standard Bangla.	Rule-based word-to-word mapping and morphological suffix transformation.	Achieved 94.75% accuracy for simple sentences but failed to handle context-dependent semantic ambiguities.

Bhattacharjee et al [14].	2023	BanglaNLG and BanglaT5: Benchmarks and Resources for Evaluating Low-Resource Natural Language Generation in Bangla	Pre-training of "BanglaT5" (Seq2Seq Transformer) on 27.5 GB of Bengali text.	Established state-of-the-art results for Bengali text generation tasks, outperforming multilingual models like mBART50.
Khandaker et al [15].	2025	Bridging Dialects: Translating Standard Bangla to Regional Variants Using Neural Models	Fine-tuning BanglaT5, mT5, and mBART50 to translate Standard Bangla into 5 regional dialects.	BanglaT5 achieved the best performance (CER 12.3%), validating its capability for dialect translation tasks.
Sultana et al [16].	2025	ONUBAD: A comprehensive dataset for automated conversion of Bangla regional dialects into standard Bengali dialect	Creation of a parallel corpus for translating regional dialects into Standard Bengali.	Provided essential ground-truth data for training NMT models to perform dialect-to-standard conversion.
Samin et al [19].	2024	BanglaDialecto: An End-to-End AI-Powered Regional Speech Standardization	Cascaded pipeline using Whisper for ASR and LLMs for translation, focused on the Noakhali dialect.	Validated the two-stage approach (ASR + NMT) but was limited to a single dialect and did not address multi-regional scalability.
Begum et al [17].	2018	Text-to-Speech Synthesis System for Mymensinghiya Dialect of Bangla Language	Concatenative speech synthesis using the Festival toolkit for the Mymensinghiya dialect.	Highlighted phonological differences in dialects but focused on speech generation rather than recognition or standardization.
Ahmed et al [18].	2019	Text to Speech Synthesis for Bangla Language	Development of a lightweight TTS system for Standard Bangla using grapheme-to-phoneme rules.	Demonstrated effective rule-based text processing for Standard Bengali, serving as a baseline for phonetic analysis.

2.3 Gap Analysis

Table *2* shows six studies on Bengali voice recognition - what they aimed for, how they worked, where they were based, and what held them back. Earlier efforts looked mostly at basic speech detection - like the Real-time Standard ASR study [20] - or zoomed in on just one regional way of speaking [1], as seen in the Dialect Standardization report [19]. The Real-time Dialect Assistant research project [8] managed turning spoken words into text yet skipped handling written forms. Over time the Standard to Dialect research was mainly focused on translation of standard language into local dialects. This did not work on actual real-time speech. Most existing tools still have numerous problems. They are slow to respond and only support a limited number of accents. They also misinterpret speech and don't succeed outside of narrow areas.

This work is different. It combines the multi-dialect speech recognition and automatic standardization. It uses Whisper Medium, BanglaT5 and Smart Fix in 10 regions. Because of this, it is able to handle more dialects than older systems. Citing the original technique, it does not neglect sudden dialect changes in the live stream audio. It is learning from these changes and still providing a stable output. The system also doesn't have complex two-step designs. Instead, it makes use of a simpler and more efficient setup, based on model compression.

Most likely, that makes it stronger when used outside labs - not just reaching further but growing easier without losing stability.

Table 2: Gap Analysis Table

Study	Core Objective	Methodology	Dialect Coverage	Key Limitation
Nasib et al [1].	Real-time Standard ASR	CMU Sphinx 4 (HMM)	Standard Bengali	High latency; Fails on dialectal/continuous speech.
Hasan et al [8].	Real-time Dialect Assistant	Fine Tuned IndicWav2Vec	10 Regions	Focuses only on Transcriptions; do not standardize text.
Samin et al [19].	Dialect Standardization	Whisper ASR+ LLM	Noakhali Region Only	Limited Scope; does not generalize to other major dialects.

Khandaker et al [15].	Standard-to-Dialect	BanglaT5, mBART50	5 Regions	Inverse task (Standard to Dialect); does not handle ASR.
Milon et al [13].	Dialect Conversion	Rule Based Mapping	Chittagong Region Only	Fails on context-dependent ambiguity; lacks ASR component.
Proposed Work	Multi-Dialect ASR+ Standardization	Whisper Medium+BanglaT5+Smart Fix	10 Different Districts	Requires two-stage inference (addressed via quantization).

2.4 Summary

The literature review of the current thesis work provides an overview of the history of Bengali Automatic Speech Recognition (ASR), starting with the first toolkit CMU Sphinx and SAPI [1], [2] and moving on to the latest models including Wav2Vec-BERT and Whisper [6]. Even though these modern models demonstrate reasonable performance when limited number of test conditions is considered here [7], they have severe shortcomings in terms of their adaptation to local phonological differences, which reduces the overall accuracy of transcription [8]. To overcome this drawback, mixed-dialect corpora, including RegSpeech12 [9] and BRADS [10] have been used, along with novel training pipelines [11], [12]. Instead, the systems such as BanglaTalk that are commercially available are in most cases phoneme-to-text translators and not contextual translators [8]. Along with it, the translation paradigm has shifted not only beyond fixed rule-based systems [13] but also to sequence-based systems such as BangaliT5 [14]. With the same development, however, there still exists a most nagging question: despite these developments, only one system is able to so far provide a holistic voice perception relationship across dialects, normalize the writing output, or stop grammatical anomalies [19], [20]. Based on this, the current research project suggests a pipeline methodology, which will start with Whisper, then with BanglaT5 and finally with a more advanced processing phase known as Smart Fix that will correct the post processing.

Chapter 3

Research Methodology

3.1 Methodology

This research is carried out using a step by step methodology. The primary objective is to work and discuss with regional Bengali speech and to be able to understand it. Firstly, the data was carefully collected and checked with attention. Then the audio and text were processed in order for the models to learn them properly. The reason we have used pretrained models is that they know about speech patterns already. After that, training and fine tuning was performed using regional data wavs. Each step was done in order to keep the dialect features in a natural such a way and without changing it. This method helps the system to do better job on the real spoken Bengali audio.

3.1.1 Overview

For this, the current study incorporates a strong two-tier architecture which connects local Bengali styles and colloquial standard variety of Bengali. Firstly, we trained the Whisper Medium Model with the "Vasha Bichitra" Dataset and get the regional transcript.Then we finetuned the BanglaT5 with the regional data. This is how we collected the material, followed its patterns and prepared it - that magic base layer of the process itself.

3.1.2 Data Collection and Description

We have collected the dataset from a competition named ভাষা-বিচিত্রা: ASR for Regional Dialects at IUT ICT Fest 2024 Datathon. The dataset has 10,674 wav audio file for training and 2668 for testing with the area of the 373 annotators from ten different districts within Bangladesh named: Rangpur, Kishoreganj, Narail, Chittagong, Narsingdi, Tangail, Habiganj, Barishal, Sylhet add one plus Sandwip. As told earlier, total audio duration is around 63 hours. On top of that, we got a transcription file featuring a regional dialect. We resampled here every audio clip in 16 kHz resampling rate to fit the audio data in the model well.

3.1.3 Data Preprocessing

The Data Preprocessing was done by us through some critical and necessary steps. Not all the regional WAV files were at 16 kHz. So, we have to convert it to 16kHz manually.

The preprocessing steps in our dataset include:

Text Normalization: We removed unnecessary texts, words, token, special signs, English character from the TSV file. We used UTF-8 based encoding here for Bangla.
Punctuation Removal: We also removed Comma, Full stops and stopping signs, special signs for better result and for better accuracy.
TSV Conversion: We converted the CSV to TSV to demonstrate and visualize the Bangla letters correctly without encoding breaking. Also, TSV data loads faster than CSV as usual. There is no chance of Encoding breaking in TSV.

3.1.4 Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed in this research so as to know more about the dataset. In this step, we have seen the distribution method of the data and the location of the differences. It made us see the discreproportion, difference, and general patterns that we see in the recordings. We were primarily concerned and aware with the distribution in the district level and sample size. With this analysis, a clear picture shows on what kind of data is to be fed to the model before training the model.

3.1.4.1 Dataset Distribution by District

The dataset was from ten different districts, it was meant to display how folks talk different places. Some regions had much more recordings in common with each other - Figure 1 brings that into focus. Leading by a big margin, Sylhet had 2,903 audio snippets. Close after it Kishoreganj brought in 1,638. Then followed Narail District with 1488; just behind Chittagong with 1406. Narsingdi had 1,098 wav files, just ahead of that of Sandwip area with 1,049. Rangpur hit 1037, Tangail reached 987 data while Habiganj has less. It is like 940 - these brought up the rear. Barishal has the less number of data. It is about 796 wavs. This difference seems to often make the group sizes too unequal to train in the model well, should things not be locked down. No shortcuts here - we didn't short-change anything (not even synthetic samples, not even adjusted penalties), even if our split wasn't balanced. OpenAI Whisper - Medium. This model is fine as it handles the regional speech styles quite well. Instead of using a single dialect, instead it trained using tons of voice clips - close to 680k hours - in lots of tongues. That way, it got the flavor of the way Bengal people speak. Of course, messing around with beats is better than trashing everything to the ground to start off from. Less samples such as Barishal (n=796) still work fine here; they manage to pack enough variety in pace, which helps to teach timing step by step and increase steadiness in the system. That you cannot predict very well - or to some extent predict - what comes next, thanks to that reliability, thanks to such setups that are better out compared to standard setups.

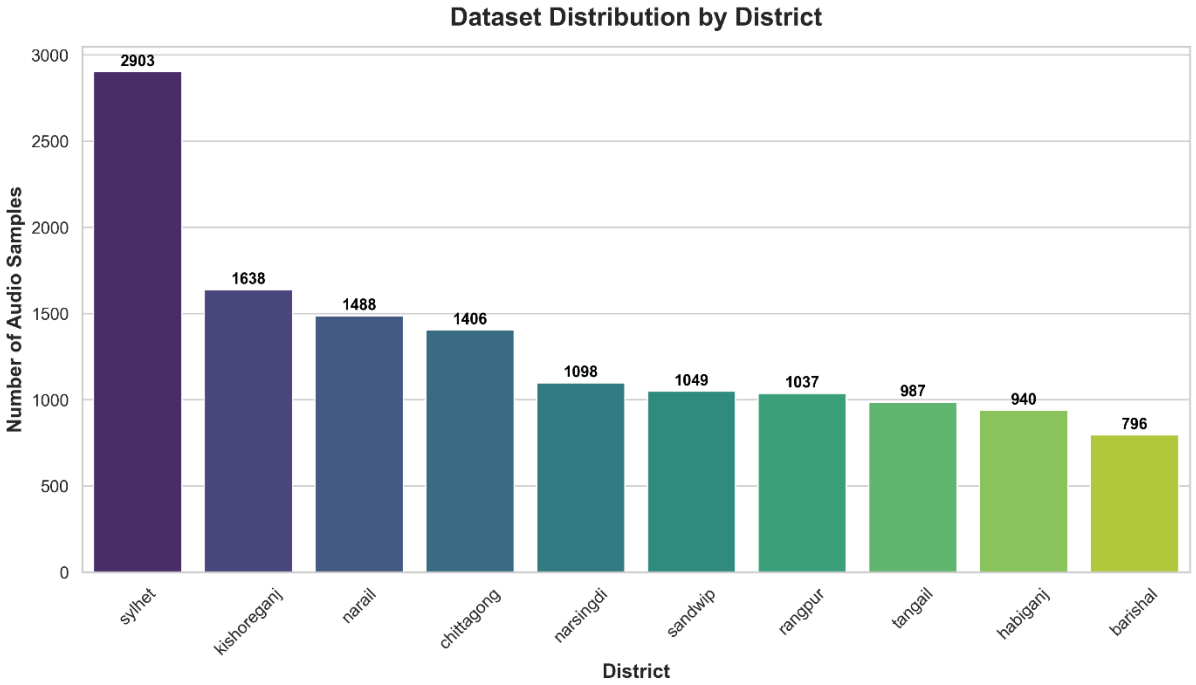


Figure 1: Quantitative distribution of audio samples across ten districts.

3.1.4.2 Sample Size Vs Vocabulary Richness

We carefully examined the effect of each dialect on the unbalanced dataset that we have collected. Having more words does not mean having more audio samples, as this is well illustrated for more words in Figure 2. For example, Habiganj has less recordings but an increase in the word count much faster than others. In comparison, Sylhet contains a lot of audio files which makes the data more stable, but it does not have the content variety as much. This is important behavior when we try to adjust the Whisper and BanglaT5 models. Minority dialects tend to include rare sounds and unique words, which become more clear in every round of training. Even in this shorter audio duration, this diversity helps to balance the data so much and to avoid overfitting of this model with respect to repeated patterns from larger groups of dataset samples.

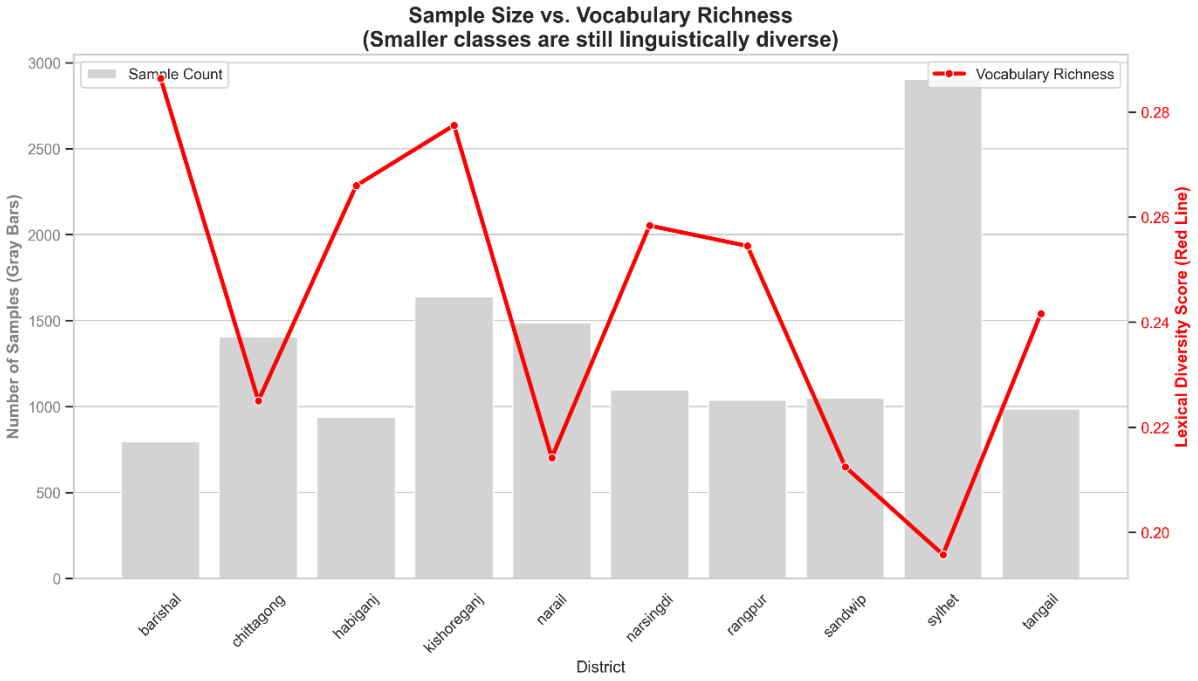


Figure 2: Comparative analysis of Sample Size versus Lexical Diversity.

3.1.4.3 Vocabulary size and Most Frequent Words

The analysis of the corpus vocabulary, which is given in Figure *3*, gives the lexical depth and the structure of the dataset. The first panel on the left shows the number of distinct types of words that are unique to each district. In line with the distribution of the sample size, the Sylhet dialect has the most vocabulary ($n=20,821), as it represents the larger amount of data. It is also interesting to note that even underrepresented dialects have a large lexical foundation; e.g. Barishal and Habiganj have 7801 and 8760 unique word types, respectively. This observation suggests that the minority classes still have enough lexical diversity to enable mapping of phoneme to graphene when the ASR model is being trained despite the imbalance in the duration of audio.

The right column mentions the top 20 most common tokens in the corpus. The distribution follows a typical Zipfian law, and is dominated by high-frequency function words, including the negation na, the pronouns “ami” and “amar”, particles “ar” and “je”. These morphemes of functional nature make the speech data naturalistic. These types of common tokens are used as linguistic anchors, which allows the model to discover common grammar structures across dialects, and focus its resources on common dialect nouns and verbs that fill the long tail of the vocabulary distribution.

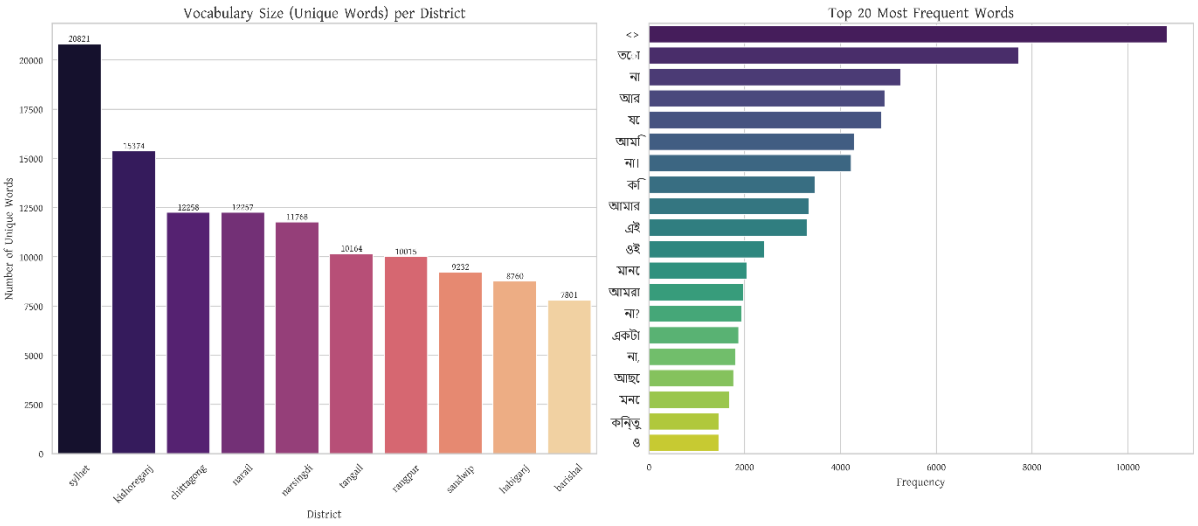


Figure 3: Lexical composition of the dialectal corpus. (Left) Absolute vocabulary size (unique word types) per district. (Right) Distribution of the top 20 most frequent tokens across the entire dataset.

3.1.4.4 Word Cloud for the Transcription of the Train dataset

The central highlight of the underlying training data is visible in the displayed word cloud (Figure *4*) namely the prevalence of the so-called lexical anchors. The dataset shows strong phonological variation in different districts, but the basic lexicon, including the pronouns as "আমরা", "আমার" and the most common verbs as "আছে", "করে", etc. is rather stable.

This kind of preponderance of a common lexicon is a beneficial basis of the suggested two-stage pipeline. Through the involvement of these high-frequency tokens as contextual anchors, the BanglaT5 model can maintain syntactic integrity at the expense of redistributing its attentional resources to the normalization of dialect-specific idiosyncrasies that only occur with infrequent lexical items and populate the long tail.



Figure 4: Word cloud visualization of the corpus vocabulary.

3.1.4.5 Distribution of the audio length and Duration Variance by District

The duration stats in Figure *5* confirm that the dataset should be a good match for the OpenAI Whisper architecture. Whisper operates on 30-second blocks of audio (or "windows"). Since the average duration per segment of the corpus is 16.85 seconds, this enables the samples to be ideally placed inside at their input window.

This period is statistically preferable for the following reasons:

- Truncating Avoidance: It is well short of the 30-second hard limit and should not lose any speech data due to cropping.
- Contextual Integrity: 17 sec is enough for full sentences or complete main clauses to capture, the semantic context necessary for BanglaT5 translation.
- Padding Efficiency: There is a need of padding to achieve the 30-second tensor, but we still have a good signal-to-noise ratio compared to very short segments (e.g. <5s).

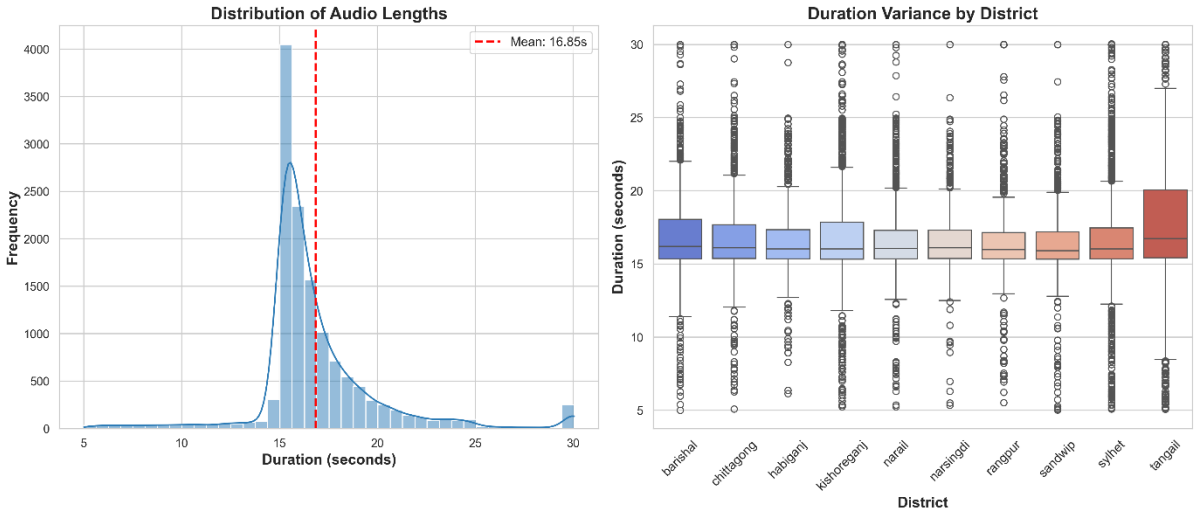


Figure 5: Temporal distribution analysis of the audio corpus. (Left) Global histogram of audio segment lengths with a kernel density estimate. (Right) Box-plot comparison of duration variance across dialectal regions.

3.1.4.6 Percentage of the silence and Fundamental Frequency Distribution

Prosodic Variations Across Dialects:

The acoustic analysis (Figure *6*) provides some clues about the prosodic features of the recorded dialects. The Silence Percentage (Left Panel) is a proportional estimate of speech rate and pause duration. The upper silence ratios are seen as Barishal and Rangpur (>30%), whereas the lower ones are found in Chittagong and Tangail (<25%), indicating the possibility of a slower temporal cadence or frequent phrase-final lengthening in natural speech with respect to these two dialects.

At the same time, from Pitch Distribution (Right Panel), we can see that spectral was almost the same in different areas so recording equipment effect was excluded. The broad "bellies" of the violin plots for Chittagong and Sylhet illustrates a peak in sample density at intermediate frequencies that implies equal representation of the vocal types.

Key Data Points:

- Global Average Silence: 27.36% (This is a very safe number, anything above 40-50% would generally be considered “bad data”).
- Barishal: The most "slow" (silence) dialect.
- Chittagong: The most “fastest” in terms of dialect (less silence).
- Pitch Range: 75-260Hz (encompasses Adult Male 85-180Hz; and Adult Female 165-255Hz).

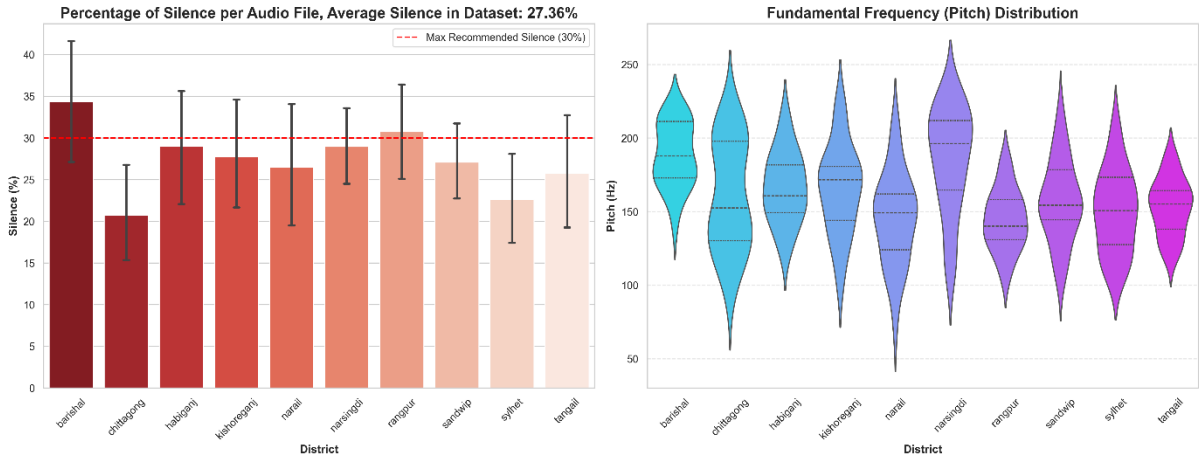


Figure 6: Acoustic feature analysis of the dialectal corpus. (Left) Mean percentage of silence per audio file with standard deviation bars. (Right) Violin plots represent the probability density of the Fundamental Frequency across districts.

3.1.4.7 Signal to Noise Ratio

An SNR analysis of the acoustic quality of the recordings shows a peaked Gaussian distribution around a high mean value (39.18 dB). As can be seen in Figure *7*, the entire data set lies safely above minimum quality necessary to support ASR, with no samples found under the 10 dB "Acceptable" level and nearly all of them surpassing the 20 dB "Excellent" point. The large number of values gathered around the 30–50 dB range also suggests a strong spectral clarity, which reinforces that recording is relatively free from environmental artifacts or background interference, something that may be influenced by could degrade the Whisper encoder's feature extraction.

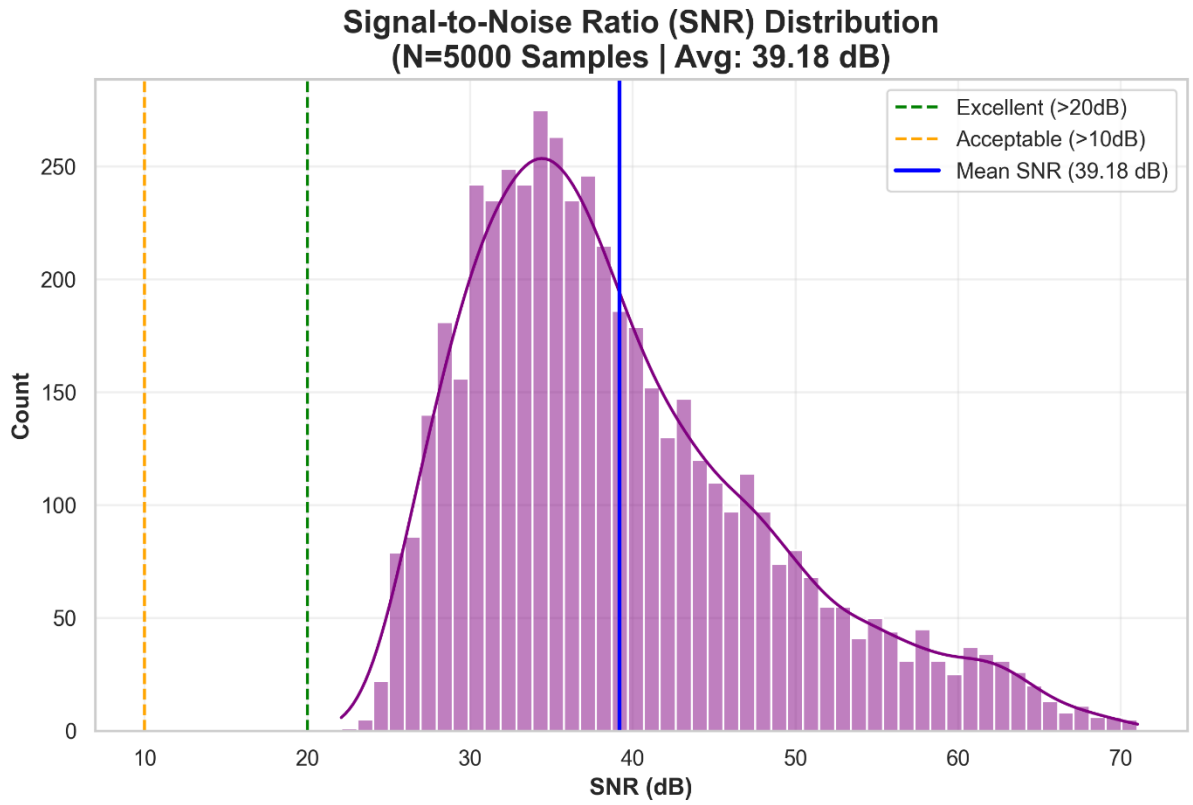


Figure 7: Distribution of Signal-to-Noise Ratio (SNR) across the dataset (N=5000 samples).

3.1.4.8 Audio Duration vs Word Count and Speaking Rate by District

We also found that Transformer-based ASR models such as Whisper are very vulnerable to cross-dialect speaking rate variation and this can destabilize attention alignment; thus, we analyzed the temporal-linguistic distribution of our corpus (Figure *8*). Results reveal a normalized speaking rate associated with a global mean of 2.25 words/s, demonstrating strong temporal homogeneity among regions. This regularity is beneficial to our model in two aspects: it stabilizes attention alignment by avoiding extreme rate differences across dialects and improves tokenization efficiency by maintaining a linear relationship between duration of an utterance and the number of tokens, thus reducing the chance of token dropping or hallucination in rapid speech or slower speech. Average district patterns confirm the presence of this uniformity: no real substructure emerges, though Tangail is most variable, followed by Chittagong, while distributions are somewhat slower over Barishal. In addition, the linearity of speech patterns observed in the scatter plot of duration vs. number of words further suggests natural pacing and not artificially constrained or irregular delivery.

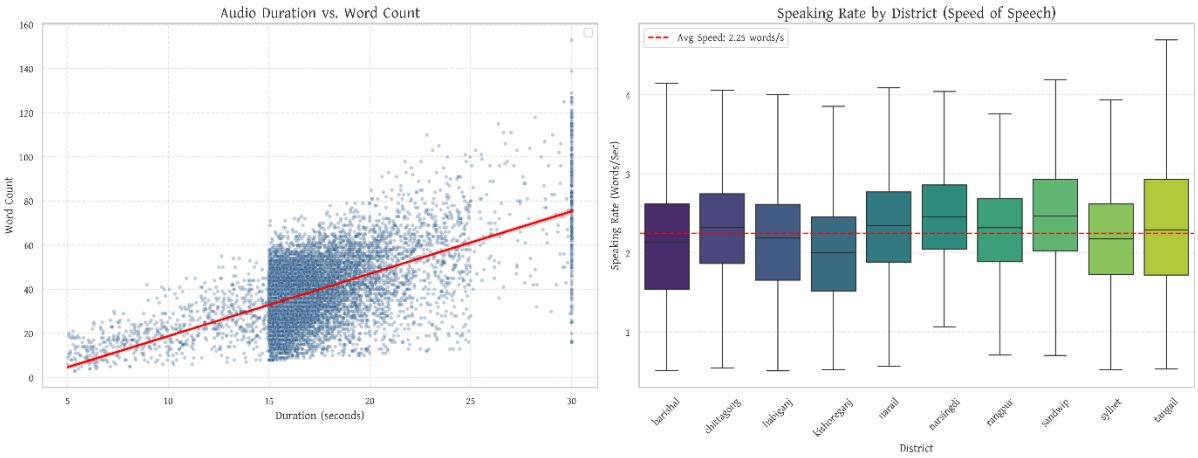


Figure 8: Analysis of temporal-linguistic correlation.

3.1.4.9 District wise Spectral Analysis

Barishal District

The spectral properties of the Barishal dialect are shown in Figure *9*. The time-domain waveform demonstrates high dynamic range speech bursts with silent portions in-between, which is indicative of an observed prosodic cadence. The Mel-spectrogram shows clear formant structures in the low frequencies corresponding to distinct phoneme articulation. The clear boundary between signal and silence facilitates the most effective preprocessing segmentation.

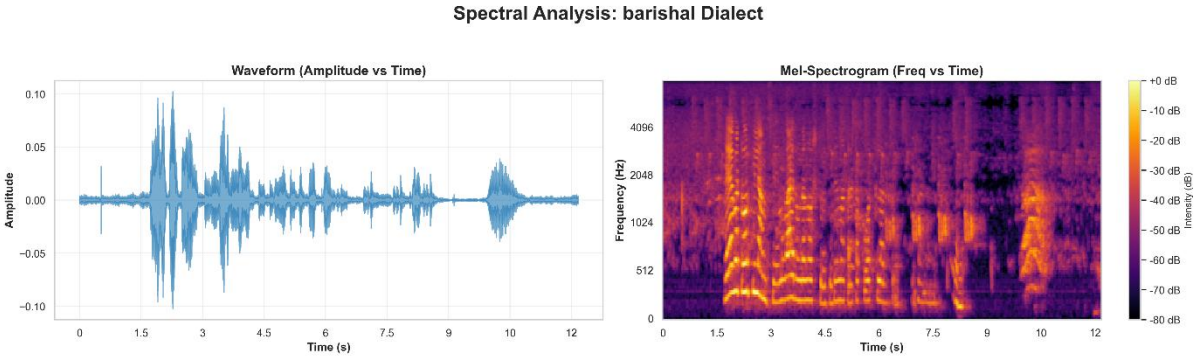


Figure 9: Time-domain waveform and Mel-spectrogram of the Barishal dialect, exhibiting distinct silence intervals and robust lower-formant structures.

Chittagong District

The acoustic feature that includes Chittagong dialect sound is as given and it can be observed through Figure *10*. The plot suggests the presence of low amplitude of all

boundaries, yet extremely few inter-words pauses (characteristic of this high rate and non-interrupted speech in this area). Its corresponding spectrogram indicates continuous energy between 1024 Hz to 4096 Hz - and this indicates a high density of the sounds forming a sudden reaction once the vowels are distorted during playback.

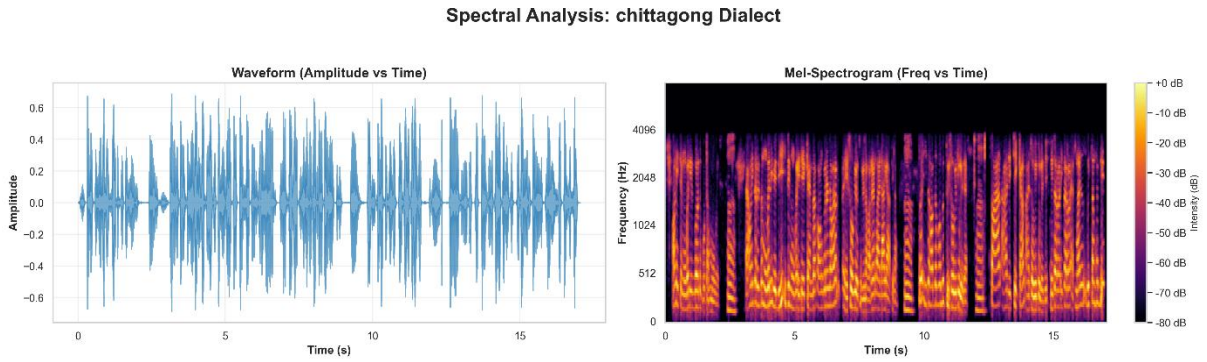


Figure 10: Listen to how Chittagong sounds - waves stay close together, energy holds firm across tones, which suggests people talk quick here.

Habiganj District

The picture here showing how does people speak in Habiganj district which is illustrated in Figure *11*. The waves have even strength and the peaks are transparent but soft and rhythmical. Middle-range sounds are very rich which makes tones easy to tell apart. Clear pitch spikes are drawn out from the background noise which is quite good for the Whisper decoder in detecting small voice's details.

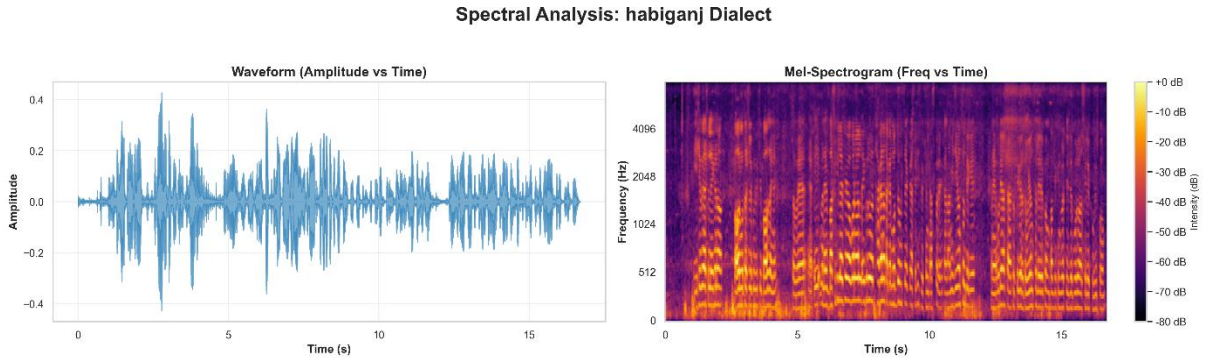


Figure 11:
Wave shape plus sound map from Habiganj speech - smooth volume shifts, distinct harmonic levels show up well.

Kishoreganj District

A sample of speech from Kishoreganj is given in figure 12. Over a period of time, the separation of different parts of the voice becomes possible, and different emphasis in syllables can be observed. The Mel-spectrogram indicates strong activity in the region of the F_0, and where there are changes in pitch and tonal shifts within that range.

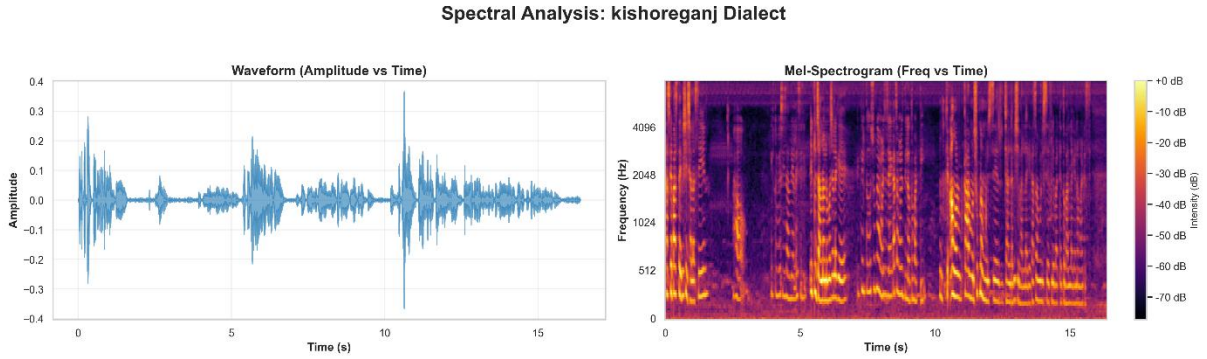


Figure 12: Spectral view of the Kishoreganj dialect, showing clear segmented speech parts and strong base-frequency peaks.

Narail District

In the Narail dialect (Figure 13) the waveform analysis shows that at onset, there are alternating high pitch bursts with onset that are followed by rapid decay. The spectral domain spectrogram representation takes advantage of clear difference between the speech activity and noise floor (dark background of the spectrogram). This difference is relevant so that phonemic boundaries and the prevention of the ASR decoder against hallucinations might be taken care of.

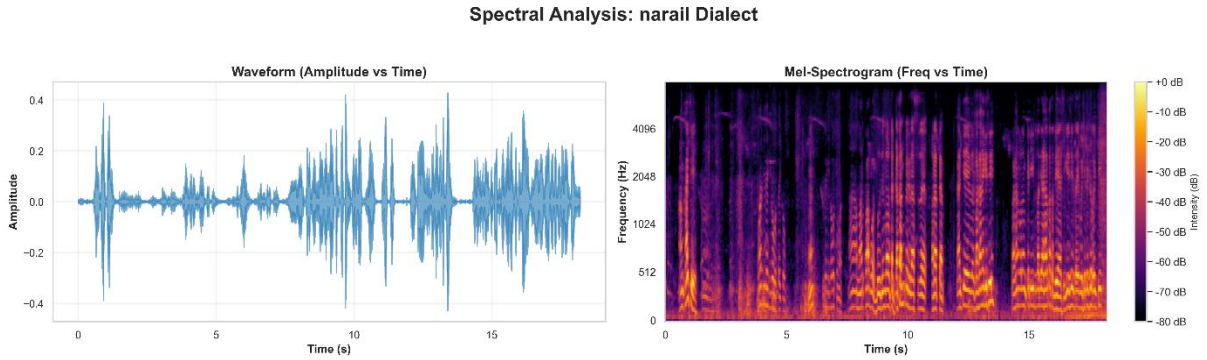


Figure 13: Time-frequency representation of the Narail dialect, showing clear separation between speech onsets and the noise floor with distinct formant tracking.

Narsingdi District

The characteristics of Narsingdi dialect are shown in Figure 14. The waveform is marked by huge amplitude fluctuation with a vigorous portion at the conclusion of the sample. The spectrogram presents high spectral density throughout the entire frequency range that indicates a greater dynamic articulatory style. The visibility of upper formants indicates that high-frequency fricatives are well-preserved in the recording.

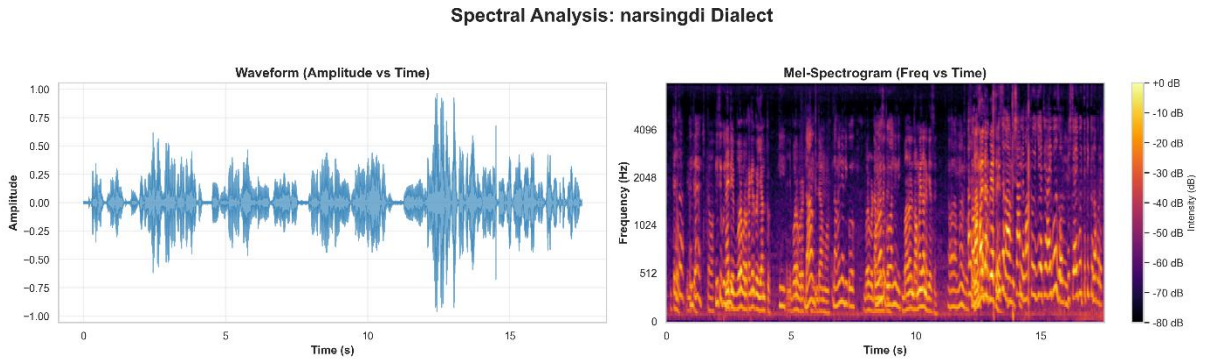


Figure 14: Acoustic features of the Narsingdi dialect, revealing significant amplitude variance and intense spectral density across the bandwidth.

Rangpur District

Figure 15 displays the spectral features of the Rangpur dialect. The time-domain signal is the signal which shows impulsive amplitude peaks which are attributable to emphatic speaking style that is characteristic of this dialect. The frequency domain analysis supports independent formant patterns that are required to explain the specific vowel changes in the dialect and differentiate them with the vowel changes in the Standard Bangla.

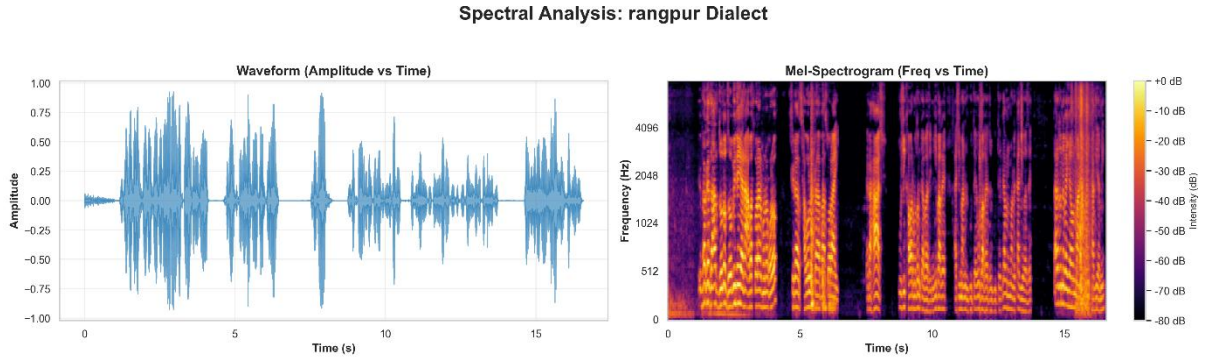


Figure 15: Waveform and Mel-spectrogram of the Rangpur dialect, characterized by impulsive amplitude spikes and distinct vowel formant trajectories.

Sandwip Area

Figure 16 represents the Sandwip dialect. The figure demonstrates the speech flow without much amplitude modulation as it is in other districts. Even the distribution of energy within the frequency bands in the higher frequency ranges that feature fricative and plosive sounds that are typical of the phonological system of the dialect is in line with Mel-spectrogram. The distinct "vertical" striations in the spectrogram correspond to clear glottal pulses.

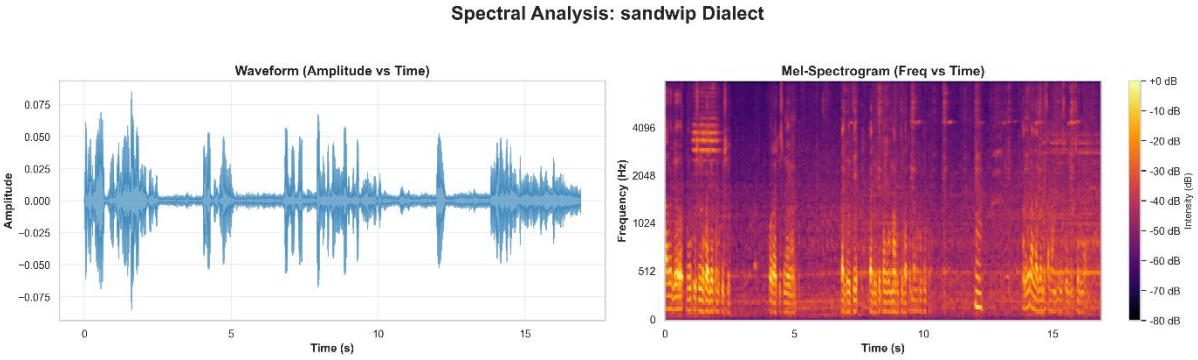


Figure 16: Spectral analysis of the Sandwip dialect, depicting a continuous amplitude envelope and consistent high-frequency energy distribution.

Sylhet District

A bit of Sylheti comes to light in Figure 17. No smooth going - jumps in occur - indicates extreme emotion or strain talking The jagged bits to tone swing fast to tight range. Besides that, newer Transformer systems perform stronger as they are able to connect the words far off than how earlier designs performed.

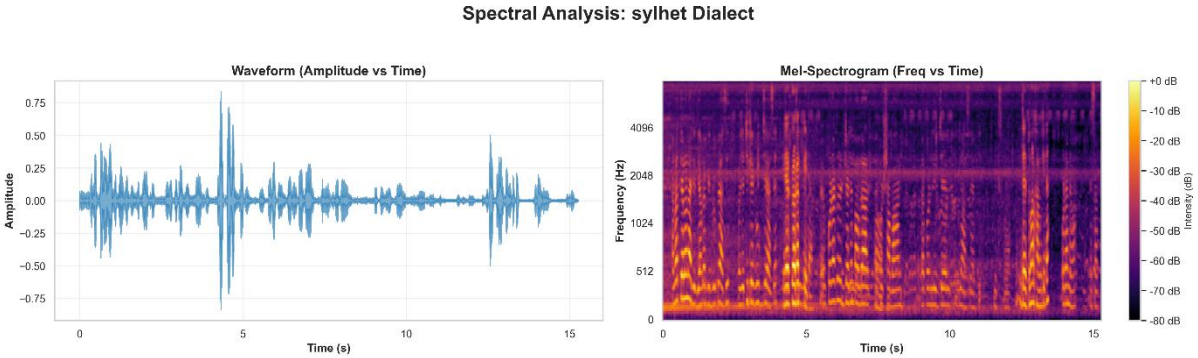


Figure 17: Acoustic profile of the Sylhet dialect, with strong waveform peaks and deep spectral lines that reflect its complex intonation.

Tangail District

Finally, a sample speech from Tangail is found in Figure 18. The rhythm is choppy, with little breaks from the other person, as normal conversation would. Instead of smooth flows there are natural gaps. There are strong harmonic patterns evident in the Mel-spectrogram. The lower formants remain stable rather than moving around.

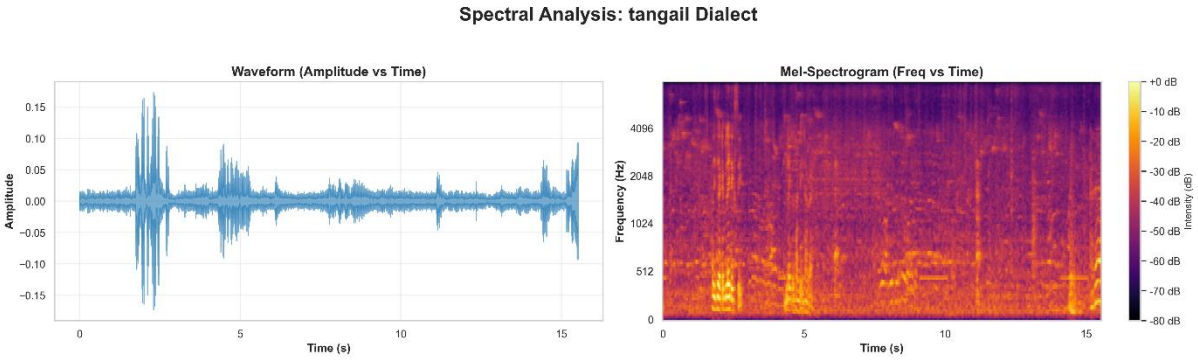


Figure 18: Time-domain and spectral visualization of the Tangail dialect, illustrating a varied prosodic contour with stable harmonic structures in voiced regions.

3.1.4.10 Log Power Spectrogram and Spectral Centroid Analysis

The *Spectral Centroid* (Ct) was calculated in order to measure the brightness and spectral center of gravity of the dialectal audio files. It is a predictor of the perceived timbre and mathematically, it is the weighted average of the component frequencies. The centroid can be computed over a certain period of time "t" as follows:

$$C_t = \frac{\sum_{n=0}^{N-1} f(n)\, x_t(n)}{\sum_{n=0}^{N-1} x_t(n)}$$

Where:

- C_t: Represents the Spectral Centroid value (in Hz) at time frame t.
- $x_t(n)$: Denotes the magnitude of the Short-Time Fourier Transform (STFT) at frequency bin n.
- f(n): Represents the center frequency corresponding to bin n.
- N: Is the total number of frequency bins (determined by the FFT window size, typically N_{FFT}/2 + 1).

Barishal District

The spectral energy distribution of the Barishal dialect is shown in Figure *19*. Log Power Spectrogram indicates that the energy of high intensity (yellow/orange areas) is concentrated in the low frequency bands that would be associated with the fundamental voicing. The spectral centroid trajectory (white line) is a constant oscillation across the spectrum 1024 Hz-2048 Hz range, which follows the voiced portions formants. This regularity represents a stable vocal resonance having clear phonetic transition.

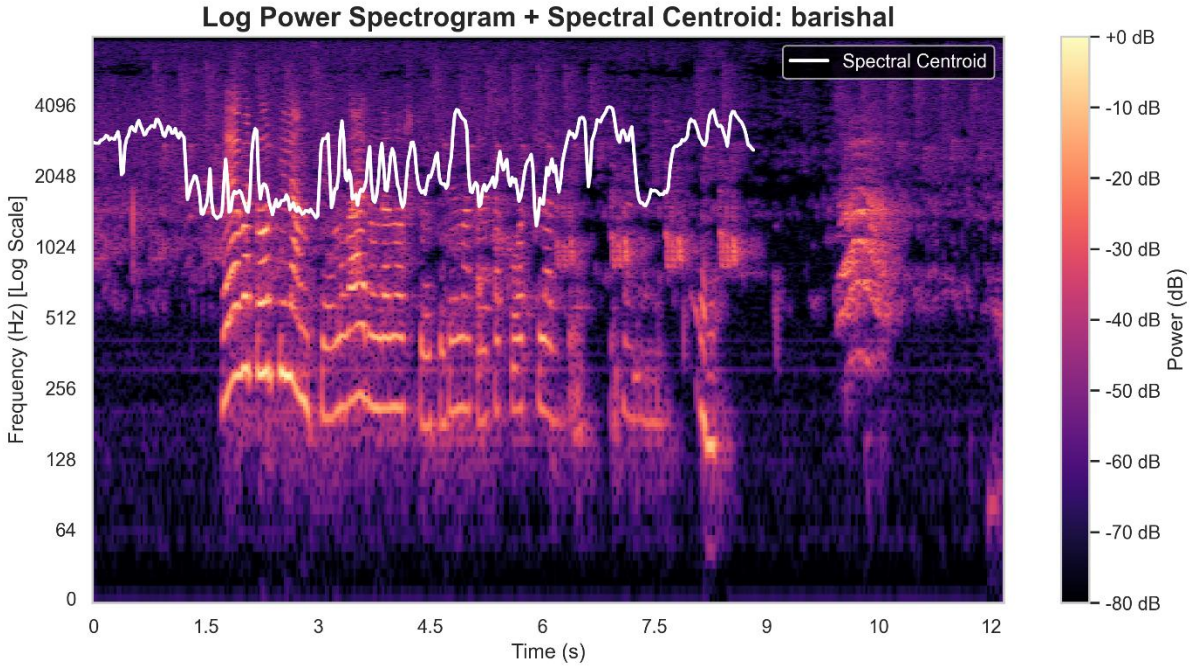


Figure 19: Spectro-temporal analysis of the Barishal dialect. The white spectral centroid trajectory (C_t) demonstrates stability during voiced segments, with energy concentrated in the lower frequency bands (<1024 Hz).

Chittagong District

Figure 20 gives the spectral profile of the Chittagong dialect. Spectrogram represents a high-density continuous energy distribution that provides an indication of the fast rate of speech that is typical of this part. The spectral centroid is very volatile where very often sharp peaks stretch up to higher frequencies. This jagged curve implies that there was an abundance of fricatives and swift articulation alternation, which also means that there was a linguistically complicated signal, but one that had high spectral brightness.

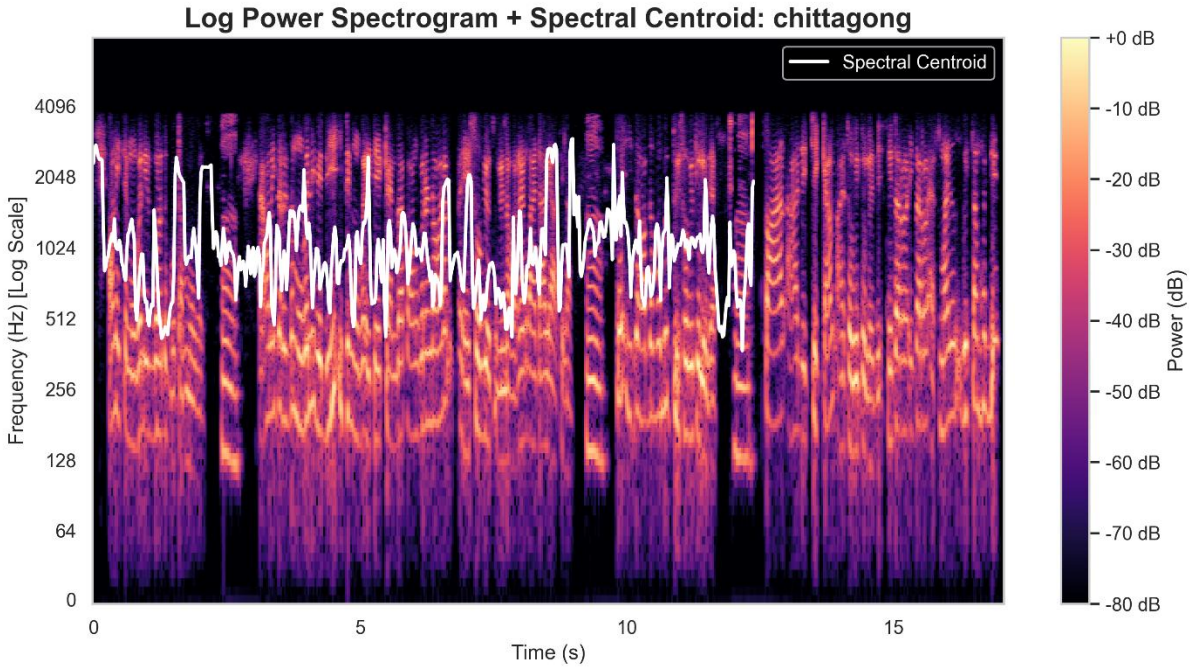


Figure 20: Log Power Spectrogram of the Chittagong dialect.

Habiganj District

The acoustic characteristics of the dialect of Habiganj are pictured in Figure 21. Spectrogram indicates harmonic stacks are well defined which denotes voiced speech. The spectral centroid has a fairly high mean value, implying a lighter timber. The curve is pretty close to the transient energy bursts, which assert that the high frequency components of the phonemes of the dialect are well maintained and as such, not confused with the noise floor.

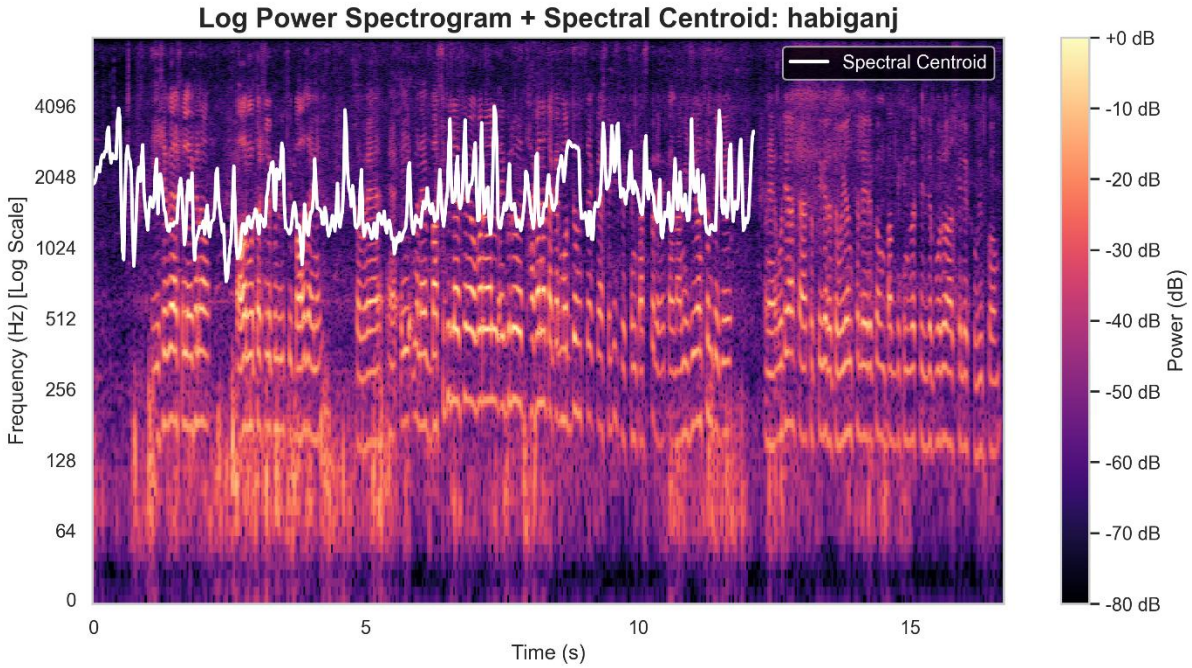


Figure 21: Spectral analysis of the Habiganj dialect.

Kishoreganj District

Figure 22 represents the analysis of Kishoreganj dialect. The spectrogram accentuates the vast compared to the small range of dynamics, the profound silences and the intense bursts of speech. The spectral centroid indeed follows these processes accurately, flat lining at their breaks, and spiking at plosive sounds. Such clear division in spectral center of gravity allows the ASR model to perform in a powerful way in the process of phoneme demarcation.

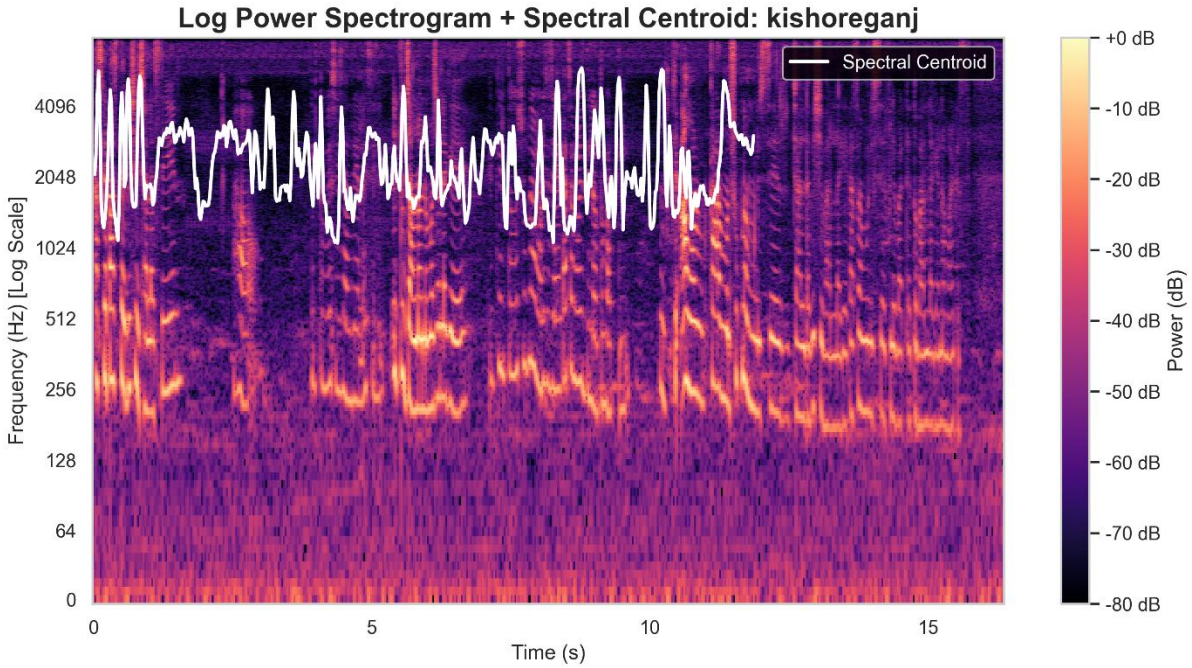


Figure 22: Time-frequency representation of the Kishoreganj dialect. The centroid trajectory accurately delineates the dynamic range between silence intervals and high-intensity plosive onsets.

Narail District

Figure 23 represents the spectral centroid result of Narail dialect. The trajectory is periodic, but rhythmic, going with the syllabic speech sample structure. The white line varies considerably meaning that there is high variation between low-frequency vowels and high-frequency consonants. This difference corroborates that the data set has the timbral richness needed in the distinction of dialects.

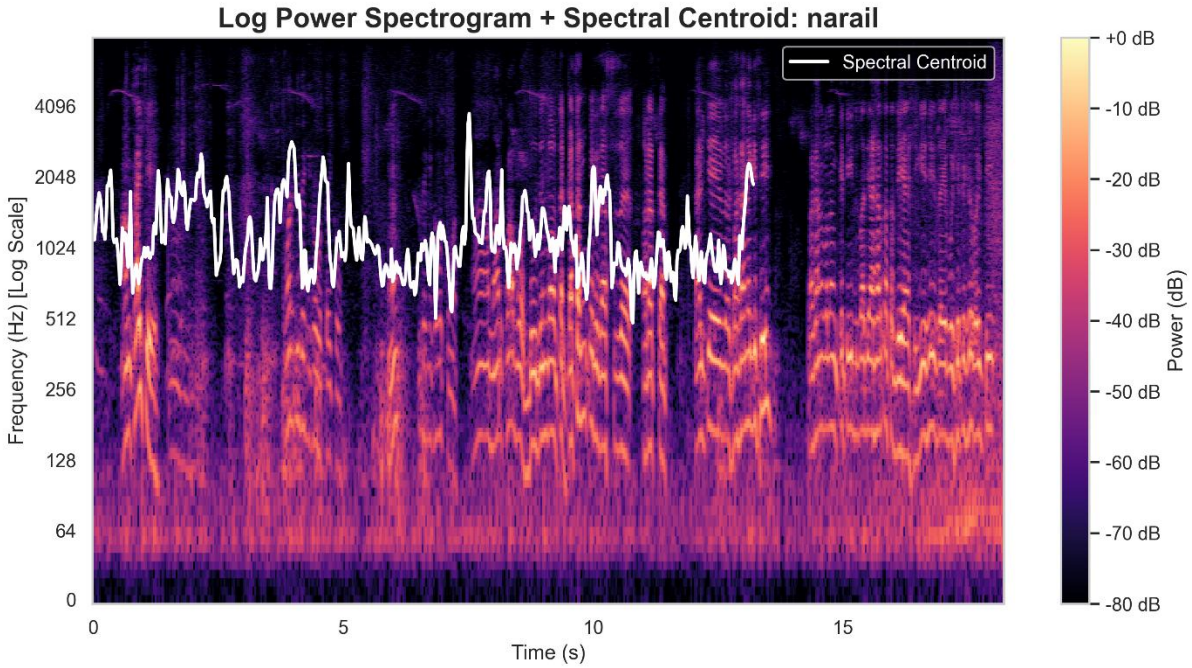


Figure 23: Spectrogram of the Narail dialect with overlaid centroid.

Narsingdi District

In Figure 24 the spectral properties of the Narsingdi dialect are seen. The spectrum energy is intense and in addition to this, the spectrogram has a lot of energy up to 4096 Hz. Spectral center of mass is also very dynamic with sharp vertical movements. This implies that the articulatory style is forceful and that there is emphasis on the high-frequency spectral content and this is a discriminative attribute of the model in the fine-tuning process.

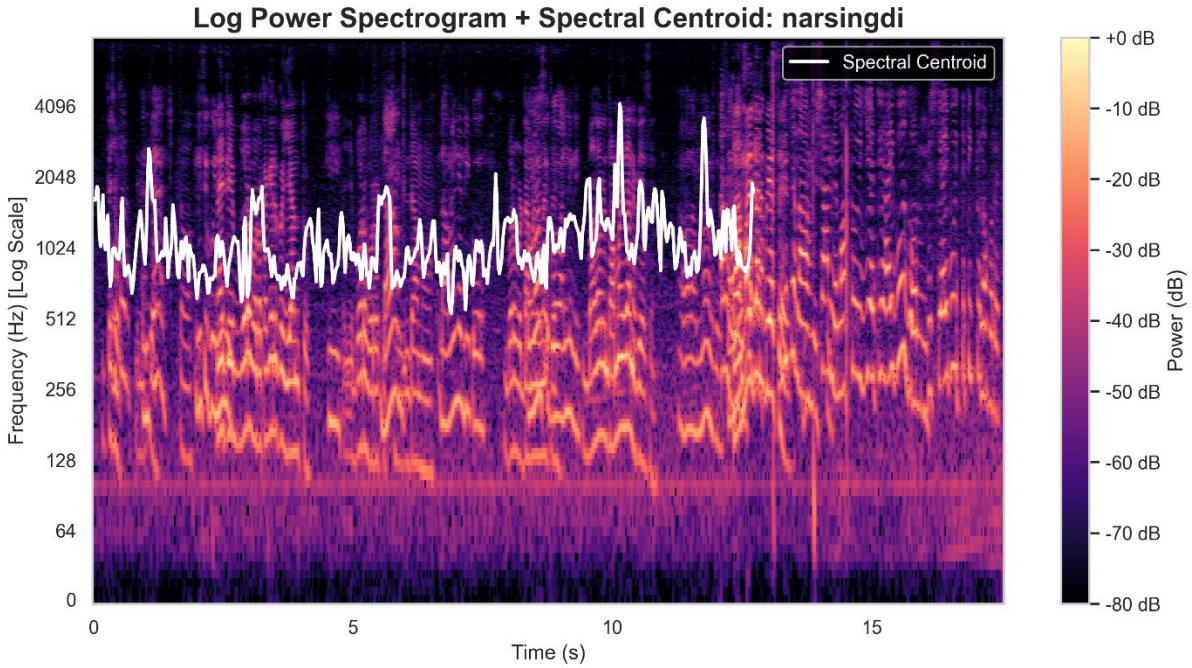


Figure 24: Acoustic analysis of the Narsingdi dialect.

Rangpur District

In Figure 25, the spectral analysis of Rangpur dialect is introduced. The white centroid line shows clear depressions in the lower frequencies bands, and this is expected given that the dialect is characterized by the high usage of rounded vowels and lower-pitch intonations. The coincidence of the centroid with the high-energy formant bands (bright vertical striations) proves that there is no spectral aliasing of the integrity of the signal.

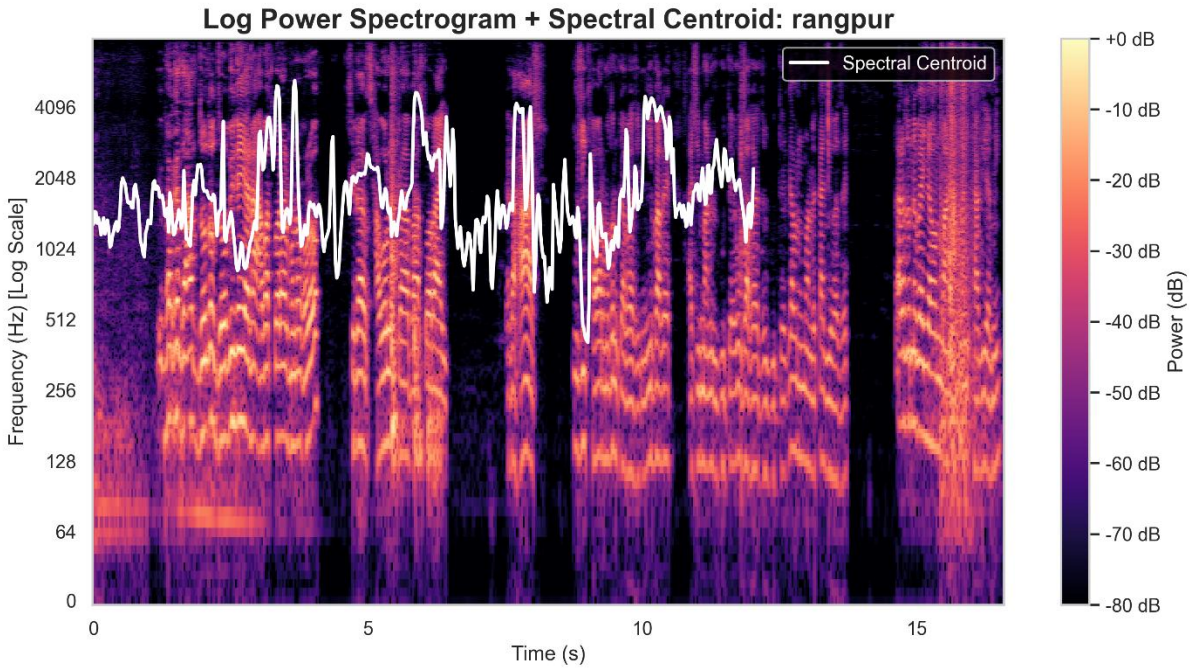


Figure 25: Spectral features of the Rangpur dialect.

Sandwip Area

Figure 26 represents the spectral properties of Sandwip dialect. The spectrogram has a wider distribution of energy in the spectrum of frequencies. The spectral centroid trajectory is a little bit wiggly, and it seizes the distinctive nature of the breathiness or aspiratory nature of the dialect that is usually characteristic of the coast dialect. This variation notwithstanding, the centroid is still identified against the background noise, and this means that linguistic features can still be extracted.

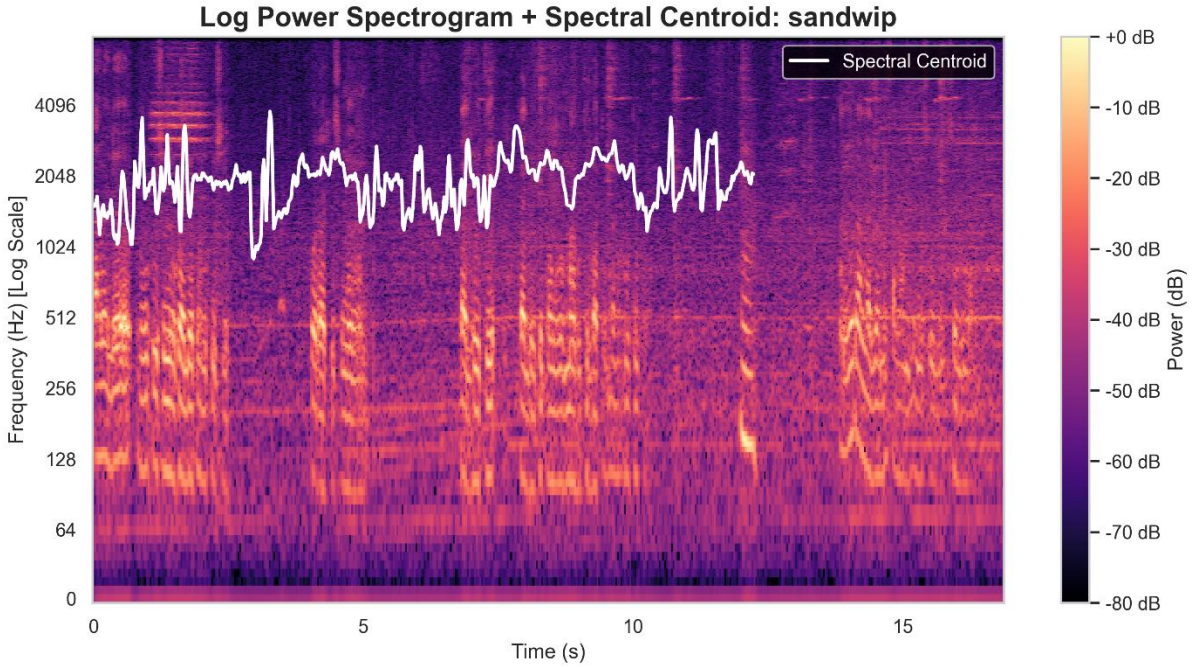


Figure 26: Log Power Spectrogram of the Sandwip dialect.

Sylhet District

Figure 27 examines the Sylhet dialect, which is complex in terms of tone. This examination is then reflected as the spectral centroid with the spectral centroid having a highly dynamic contour that fails to rest on the average frequency. The trajectory reflects the less obvious changes in pitch and timbre (tone) that makes Sylheti and Standard Bangali different, and proves that the model must pay attention to these particular changes in the spectrum.

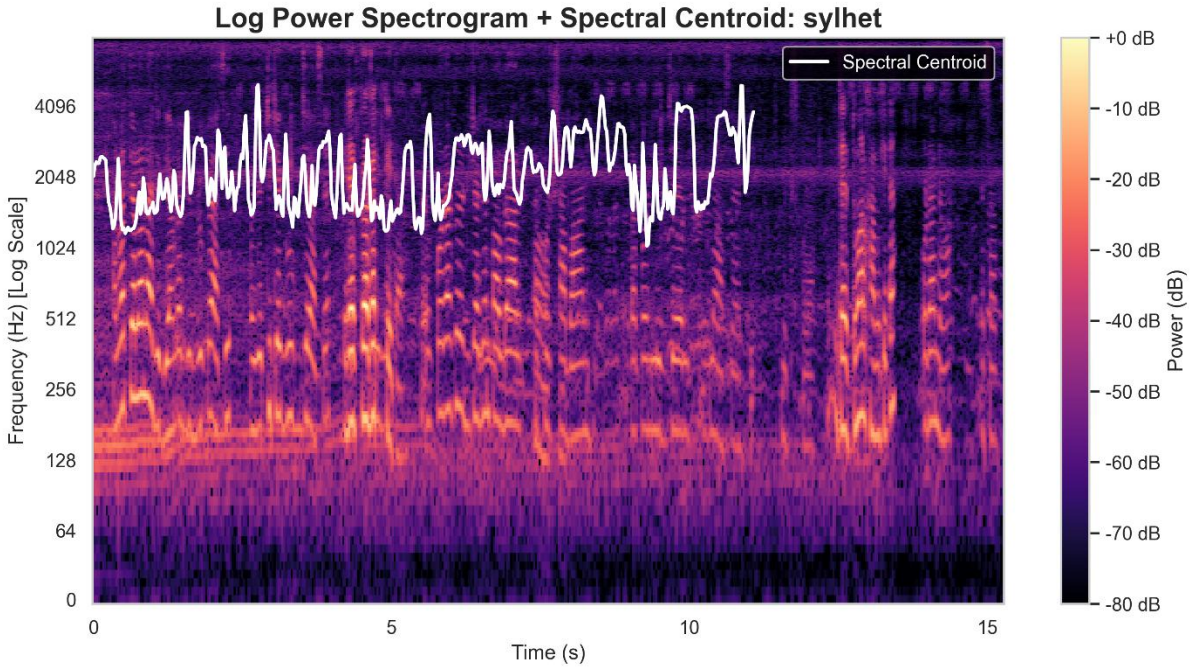


Figure 27: Spectro-temporal analysis of the Sylhet dialect.

Tangail District

Lastly Figure 28 illustrates the spectral centroid of Tangail dialect. It was clear in the plot that the brightness of the specters correlates well with the amplitude peaks. The centroid path is comparatively smooth in the middle of the vowels and exhibits acute discontinuities on the consonantal boundaries. It is this structural purity in the spectral domain that hints at the fact that the samples of Tangail have a high level of intelligibility and good acoustic qualities of automatic transcription.

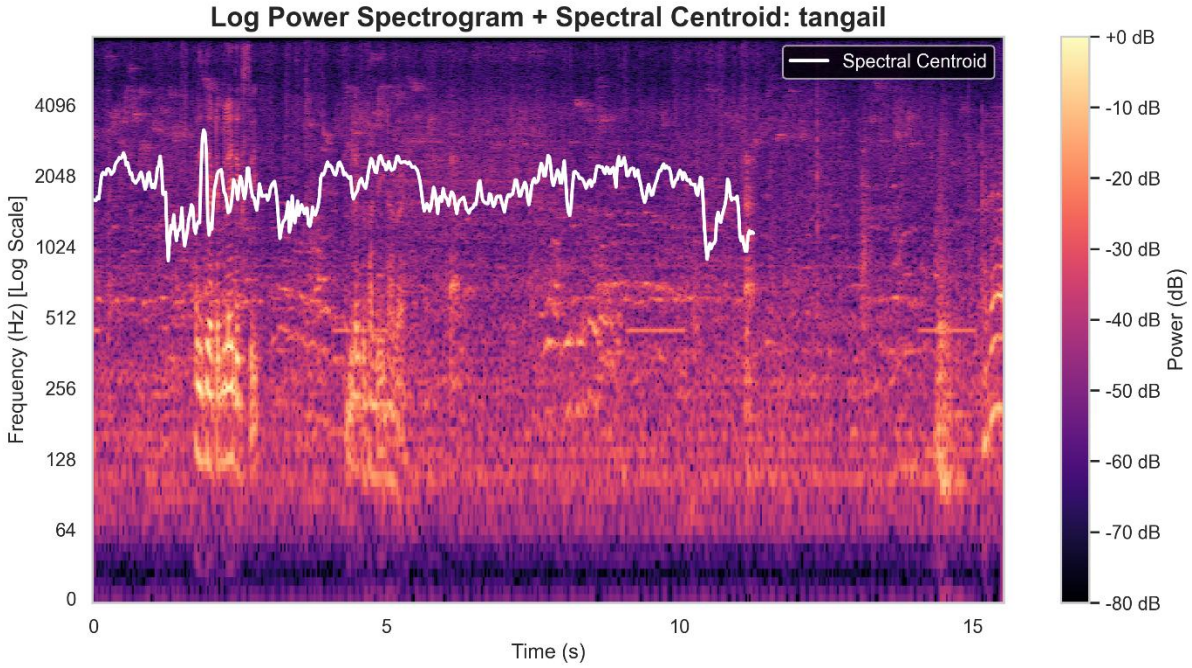


Figure 28: Spectral analysis of the Tangail dialect.

3.1.5 Train-Test Duration analysis

It is observed in the stratification check that when the data is split into train and test, the stats remain the same on each side as illustrated in Figure 29.According to this boxplot, the short or long videos (<5 s or >25 s) appear equally frequent in both train and test and therefore this edge case is viewed by the model as it is tested or as it is being trained.Mean lengths are nearly identical, 16.9s on train and 17.2s on test, a difference of 0.3s, and this is the same as padding and batching in inference is a reflection of training, thus, we minimize length bias.There was also a slight peak at 30s, indicating the presence of the longest, hardest clips in both sets hence we do not simply leave out hard examples and give the model an easier task.

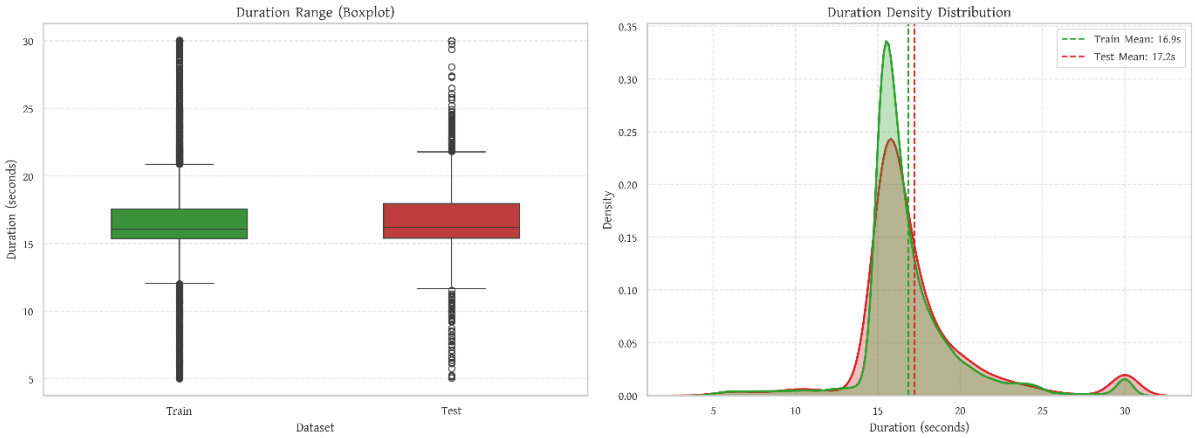


Figure 29: Comparative analysis of temporal distributions between Training and Testing partitions.

3.1.6 Proposed Methodology

The used methodology will deal with regional audio of Bengali architecture in a systematic way. It involves a process of two stage learning as the speech and text are processed independently. First, the system is train with the dialectal audio comprehension. Then it operates on changing the regional text into standardized form of Bengali. This method is used to maintain the meaning of dialect and enhance readability and accuracy of the model. The general structure is basic, efficient, and very appropriate in real world application.

3.1.6.1 System Overview

Figure *30* presents overall Architecture, the architecture of system can be divided as the two successive training stages:

Whisper Fine-tuning (Acoustic-to-Semantic Mapping): It begins with TSV and Regional

Wav Files as input to the tool. These pairs are then input into the Whisper Medium architecture (indicated by the configuration monitor icon). The model is fine-tuned to capture the specific variation in dialectal acoustics, forming a “Generated Model” that can produce accurate RTTs on unseen audio.

Semantic-to-Syntactic Normalization (BanglaT5 Fine-tuning): The second phase takes input the output from ASR module. The text in the dialect (e.g., "মুই কাইল বাড়িত যাম"), being in regional language is taken as source sequence. This data is employed to finetune BanglaT5 architecture (represented by the neural chip icon). The model is trained to align these dialectal variants to their standard counterparts (e.g., "আমি কালকে বাড়ি যাব") when appropriate, and at times normalizes grammar and vocabulary.

Validation and Output: The last module, Checking Validity consists of linguistic metrics (BLEU/WER) that are used to validate the translation by matching it with provided Standard Text references.

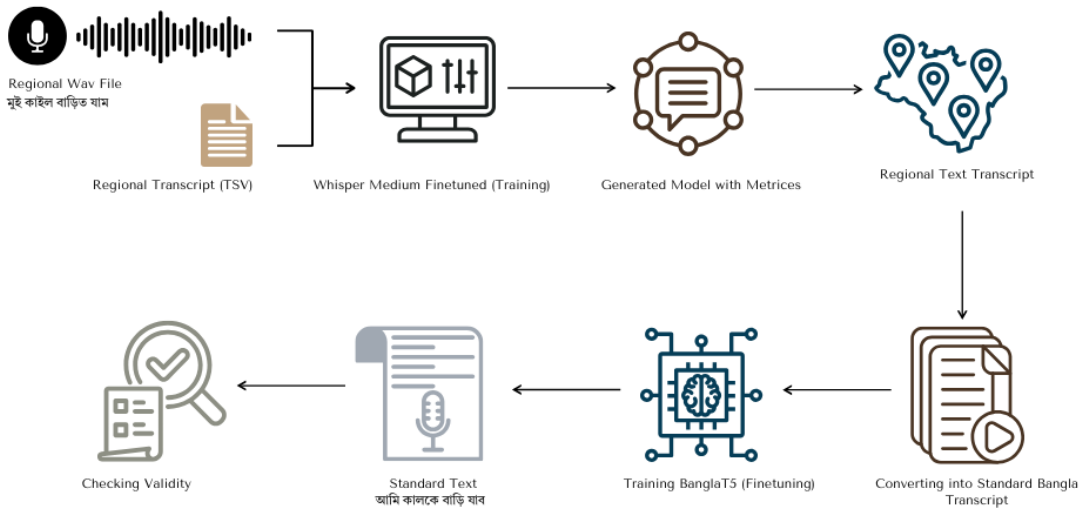


Figure 30: Schematic workflow of the proposed two-stage dialect conversion system.

3.1.6.2 Data Preprocessing Pipeline

The first phase of information processing forms the base on which our dual stage architecture is based. The pipeline takes in two major streams of data, one being raw regional WAV files and the other being ground-truth regional transcripts, as shown in the first step of Figure *30*.

- **Audio Normalization:** To ensure that all incoming audio has been compatible with the Whisper architecture, we resample all audio to 16kHz. We also do coherent

down-mixing of the stereo channels to mono-channel waveforms, thereby minimizing the computational dimensionality and retaining the important phonological information.

- **Text Formatting:** The textual information are organized in the form of a table-separated value (TSV) such that each audio file is correlated with a dialectal transcription of that audio file in a one-to-one manner. This correspondence achieves a very good level of fidelity when comparing the graphemic series to the acoustic signal at the data-loading stage.
- **Data Partitioning:** After preprocessing, the corpus is partitioned into training and test sets, thus ensuring that the amount of dialectal variation in each of the two subsets is consistent, as discussed in the Data Analysis chapter.

3.1.6.3 Module I: Dialectal ASR (OpenAI Whisper-Medium)

The initial computational module aims at modeling the Acoustic-to-Semantic gap. As the main ASR engine, we use the OpenAI Whisper (Medium) architecture which is a weakly-supervised Transformer model. As shown in Figure *30*, this module takes the pre-processed regional audio as input and produces a Regional Text Transcript. The selection of the "Medium" number of parameters strikes a balance between computational efficiency and ability to model complex dialect phonetics, which is lacking in standard Bengali ASR systems.

3.1.6.4 Architecture Adaptation

This research does not go into training a speech recognition model from the scratch as that would mean preparing thousands of hours of transcribed audio files – instead, we leverage the concept of Transfer Learning. It is fine-tuned on the language of a specific dialectal corpus, using the pre-trained multilingual Whisper model. In this stage, the internal weights of the model are tuned to reduce the difference between predicted tokens and ground TSV transcripts. This adaptation enables the model to learn dialect-specific variation in prosody and phonemes (e.g., recognizing the Sylheti fricative or Chittagonian rapid cadence), maintaining its existing knowledge of Bengali script.

3.1.6.5 Module II: Text Standardization (BanglaT5)

The second phase of the pipeline concerns it with Semantic-to-Syntactic normalization. The generated dialectal text output of Module I (e.g., "মুই কাইল বাড়িত যাম") is considered as the source sequence in a machine translation problem. This model is based on the pre-trained BanglaT5, which is a Sequence-to-Sequence (Seq2Seq) Transformer trained on large corpus of Bengali text. The workflow diagram (Figure *30*) illustrates how the regional transcript is passed to the BanglaT5 model and fine-tuned by working towards translating dialectal morphology to Standard Colloquial Bengali (SCB) (e.g., "আমি

কালকে বাড়ি যাব"). This is used to eliminate grammatical variation and standardize word usage without losing the meaning of the utterance.

3.1.6.6 Sequence-to-Sequence Formulation

The process of standardization is mathematically simulated as a conditional probability problem. Let {X= x1, x2, x3 ,……., xn} denote the set of dialectal tokens produced by the ASR module. The BanglaT5 model aims at producing a target sequence, Y= {y1, y2, y3……., ym} (Standard Bengali) such that the conditional probability $P(Y|X)$ is maximized.

$$P(Y|X) = \prod_{t=1}^{m} P(y_t|y_{<t,}X)$$

In this case, $y_{<t}$ is the history of the tokens that were made. The model learns the mapping function that maps certain dialectal markers (like "কাইল") to their standard equivalents (like "কালকে") in a way that makes the dialect conversion like a "low-resource translation" task.

3.1.6.7 Experimental Setup and Hyperparameters

The training plan was made to make sure that the models converged and didn't get overfitted at the dialectal datasets.

3.1.6.8 Computational Environment

To meet the high memory needs of Transformer-based architecture, all model fine-tuning and inference tasks were done in a high-performance computing environment.

- Hardware: NVIDIA GEFORCE RTX 4060.
- Platform: Visual Studio Code, Hugging Face, Google Colab.
- Frameworks: PyTorch, Hugging Face transformers, and datasets libraries.

3.1.6.9 Training Configuration

The hyperparameters were optimized empirically. The specific configurations for the two modules are as follows as Table 3 .

Table 3: Training Configuration of Whisper and BanglaT5

Hyperparameter	Module I: Whisper (ASR)	Module II: BanglaT5 (Standardization)
Learning Rate	$3.8\times10^{-7} \rightarrow 8.8\times10^{-7}$ (warmup)	~3×10^{-5}
Batch Size	16	16
Optimizer	AdamW	AdamW
Precision	FP32	FP32
Max Steps / Epochs	74,074 steps (≈3 epochs)	10 Epochs
Gradient Accumulation	2	1
Architecture	WhisperForConditionalGeneration	T5ForConditionalGeneration
Best Checkpoint	checkpoint-72500	checkpoint-6480
Activation Function	GELU	gated-gelu (gelu_new)

3.1.6.10 Evaluation Metrics

To check how well the system worked, we calculated quantitative metrics for both stages:

Word Error Rate (WER): This is how we check the ASR module by looking at the edit distance between the predicted dialectal text and the ground truth TSV.

$$w = \frac{S + D + I}{N}$$

BLEU Score (Bilingual Evaluation Understudy): This is how we check the BanglaT5 module by looking at how many n-grams are the same in the generated standard text and the reference standard text.

Character Error Rate (CER): An extra measure for the ASR module to take into account small spelling mistakes in Bengali script.

3.1.7 Functional and Nonfunctional Requirements

The requirements were divided into behavioral (functional) and quality (non-functional). They are:

Functional Requirements:

This is a list of what the system needs to do:

Audio Ingestion: The web dashboard will take input any type of audio file format. Also, there is a drag and drop option.

Dialect Transcription (ASR): Fine-tuned Whisper will take local dialectal speech WAVs as input and give dialect aware regional text as output.

Text Normalization: The model would be required to normalize the dialectal text with the help of BanglaT5 so that it becomes Standard Bangla transcript.

Dual Output Display: Intermediate Output (Dialectal Text) and the Final Response (Standard Text), must be displayed simultaneously for ensuring what they intend is correct.

Error Handling: We handled errors here smartly. If the audio length is too big or too small for the model to process, a popup will show.

Non-functional Requirements:

This is the list of the non-functional requirements:

Inference Latency: Latency of the audio processing is very less. Even a 15-second-long audio chunk will process very fast.

Accuracy Threshold: We must robust the system and at the same time we must maintain consistent accuracy.

Scalability: In the model, we can add a dialect of other districts anytime.

Usability: Very user friendly and simple UI so that users can easily record voice or drag and drop an audio file.

3.1.8 Data Flow Diagram

A rough drawing for DFD Level 1 - shows how data moves step by step through the target setup (check out Figure *31*). Instead of labeling parts, it tracks how raw noise turns into clear results. At the same time, this is where every element starts and finishes. Raw audio goes from the user straight to the app's starting point. Next, muffled sound moves into Preprocessing - there, it gets adjusted to 16kHz while being split into chunks. From there, Whisper medium grabs every segment, turning speech into plain text using the native language. Rather than ending now, the output goes through BanglaT5, fixing errors and meaning. Finally, clear Standard Bangla Text reaches the Output Interface where users check results, closing the loop.

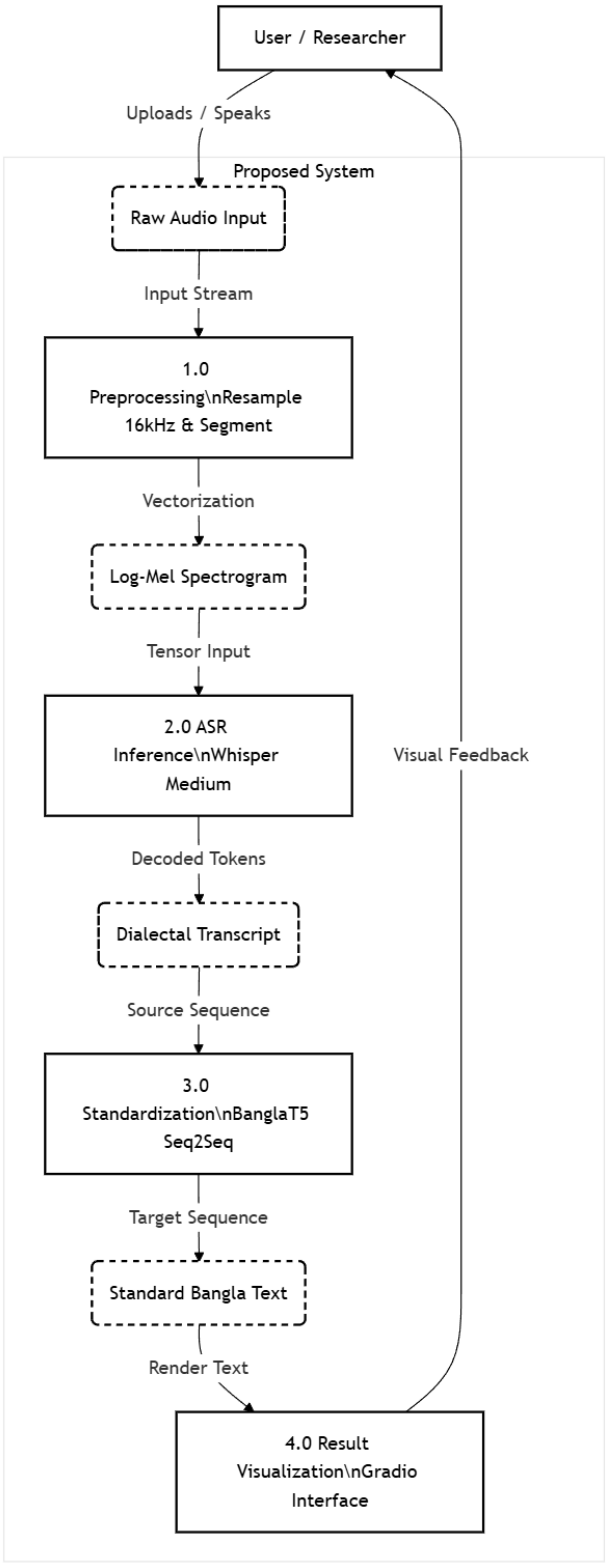


Figure 31: Data Flow Diagram of the System

3.1.9 UI Design

Because it has few dependencies and allows us to quickly prototype machine learning models, we decided to build the User Interface (UI) using the Gradio framework.

Input Section: A large "Microphone" toggle allows you to begin recording immediately, and a "Drag-and-Drop" area allows to upload files.

Control Mechanism: An independent "Generate" button initiates the inference pipeline.

Output Section: There are two divided panes in the interface. Module I's "Detected Dialect Transcript" is displayed on the left, while Module II's "Converted Standard Bengali" is displayed on the right. After ensuring that the ASR accurately heard the dialect, people can verify the translation thanks to this clear design.

Figure *32* represents the User interface of deployment.

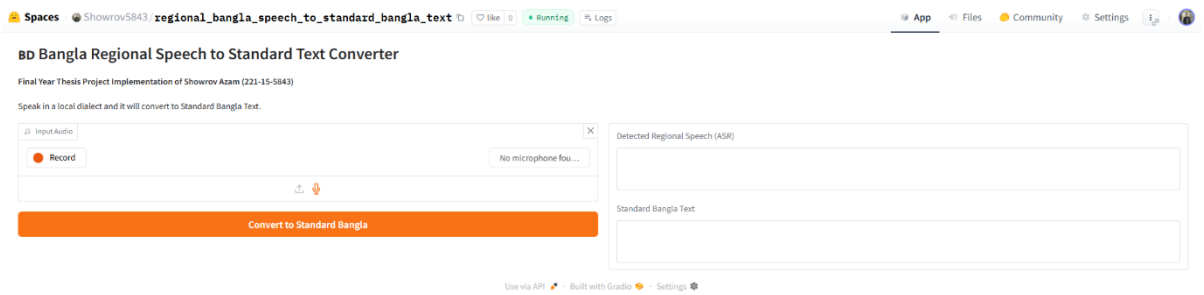


Figure 32: Graphical Interface of the UI

3.2 Detailed Methodology and Design

This section talks about why the pipeline's architecture was chosen by comparing the chosen models to other models that could have been used.

Option A: Wav2Vec 2.0/Wav2Vec XLS-R 300m (For ASR)

Description: A self-supervised framework made by Meta AI that learns how to turn raw audio into speech.

Reason for Rejection: Wav2Vec 2.0 is very powerful, but it needs a lot of fine-tuning data to work well with dialects that don't have a lot of resources. It also doesn't have the huge multilingual pre-training that Whisper does. This makes it less able to handle the "code-switching" that is common in Bengali dialects.

Alternative B: LSTM Based on Rules (To Make Things the Same)

Description: Using Long Short-Term Memory (LSTM) networks and hard-coded language rules to match words from different dialects with standard words.

Why it was turned down: Bengali dialects have more than just word swaps; they also have complicated changes in syntax. Rule-based systems can't handle these differences because they are too rigid, and LSTMs have trouble with long-range dependencies in sentences that are hard to understand.

The reason for choosing Whisper medium and BanglaT5 for the thesis work:

1. Data Efficiency (Whisper): Whisper medium default training used 680000 hours of audio data. So, it is already very well to detect Bangla Standard Language. But The model is finetuned with the dataset used here to get better accuracy and less WER.

2. Contextual Awareness (BanglaT5): BanglaT5 understands the semantic meaning and give most accurate standard Bangla transcriptions

3.3 Project Plan

Phase I: Discovery (Weeks 1–5). In this section, we do a lot of research based on previous work and find the gaps.

Phase II: Data Work (Weeks 6–10): We collected the data, preprocess the data, visualize the important parts of the dataset and remove animalities from the dataset.

Phase III: Building the Models (Weeks 11–14). In this part, we analyzed the data also, then applied the best model to get better accuracy.

Phase IV: Check-up (weeks 15 to 19). Evaluate the model, check the accuracy and other related metrices.

3.4 Task Allocation

This Table 4 gives an idea of the weeks of each task.

.

Table 4: Task Allocation Table

Tasks	Weeks																		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Discovery																			
Data Engineering																			
Model Development																			
Evaluation																			

3.5 Summary

This work examines the issues of separating regional dialects of Bangladeshi from general Bengali. It relies on a step by step neural set up rather than combining large models. First, the audio is cleaned up, downsampled to 16 kHz and everything is divided into small chunks that can be handled by the Transformers. The system then divides into two sections. Whisper Medium performs ASR and also captures tricky sounds well, including Chittagonian Rushes or Sylheti rhythm Next, BanglaT5 relocates dialect phrases to clear Standard Bengali language along with fixes the unusual spellings (they always keep the meaning). For testing two pipelines were run (one feeding word accuracy and the other one checking dialect conversion), keeping the workload light but allowing the system to be correctly evaluated.

Chapter 4

Implementation and Results

4.1 Environment Setup

Below are the exact hardware and software settings:

- Hardware Acceleration: NVIDIA Geforce RTX 4060 GPU (8GB GDDR6 VRAM).
- Processor: Intel Core i5 14400, 14th Gen Raptor Lake Processor.
- System RAM: 16 GB.

Software Framework:
Language: Python 3.9.13
Deep Learning Library: PyTorch 2.1.0+cu118
Model Libraries: Hugging Face transformers (v4.35), datasets (v2.14), and accelerate (v0.24).
Audio Processing: librosa and ffmpeg for 16kHz resampling.
Interface: Gradio (v4.0) for the web-based demonstration UI.

4.2 Testing and Evaluation Metrics

The Word Error Rate (WER) is the main measure for Module I (Whisper). The minimum number of changes needed to turn the recognized output into the reference transcript is what WER uses to measure speech recognition performance.

$$WER = \frac{S + D + I}{N} \cdot 100\%$$

Where:
S is the number of Substitutions.
D is the number of Deletions.
I is the number of Insertions.
N is the total number of words in the reference.

We used the BLEU (Bilingual Evaluation Understudy) score and the WER to rate Module II (BanglaT5).

BLEU: This measures the geometric average of n-gram precisions between the generated

and ground truth Standard Bengali text. A lower BLEU score is an indication of dissimilar textual meaning and structure.

Validation Loss: We monitored cross-entropy loss to ensure the model was converging and wasn't overfitting.

4.3 Results and Discussion

This paragraph section discuss how the proposed system performs on the post training basis. It talks about the learning process of the model with time and how the results get altered at a particular time being. The model behavior is perceived out of different evaluation scores. The analysis of the results is to understand whether the system is appropriate to manage regional speech in Bengali. The discussion of observations made based on the figures and metrics also takes place in this result and discussion section.

4.3.1 Module I: Whisper Medium (ASR) Performance

More than 20,000 steps trained the OpenAI Whisper Medium model worldwide. As seen in Figure *33*, training behavior links loss during learning with word error rates on validation. A smooth drop in training mistakes appears as a blue curve. Red dots mark actual performance checks now and then.

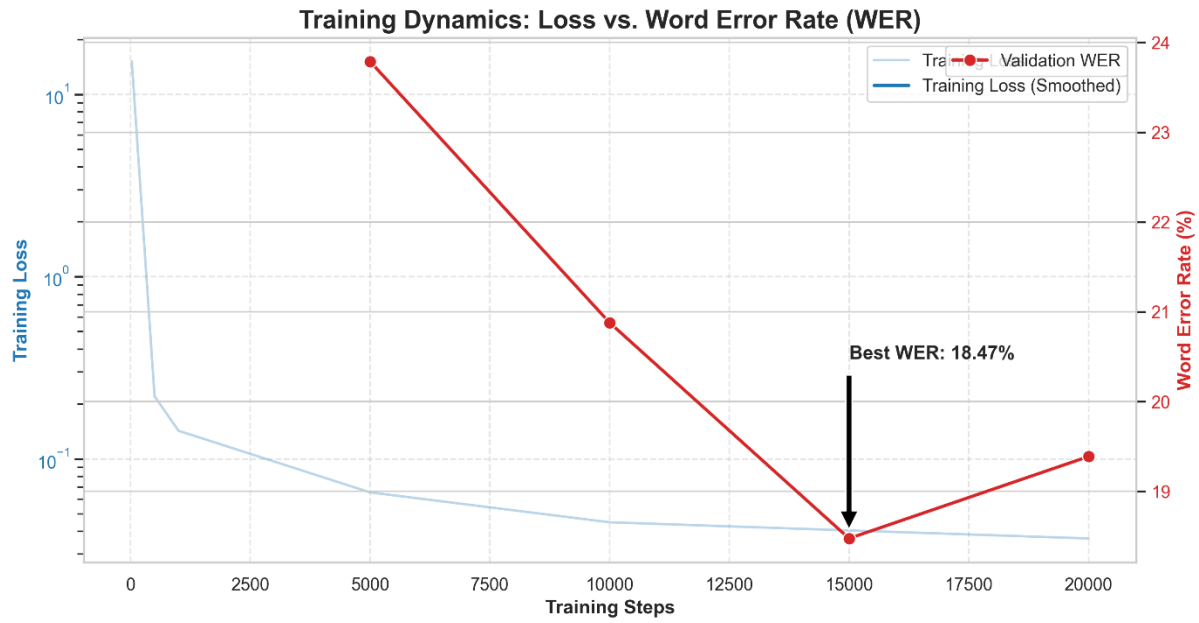


Figure 33: Training dynamics of the Whisper Medium model.

Results Analysis:

Convergence: The training loss - shown in blue - dropped quickly at first, starting above 10^1 then falling near 0.04 by step 20k. That drop means the model adjusted its earlier settings to better match how this specific speech sounds.

Best Performance: The results data show that the model's Best Metric (WER) was 18.47% at checkpoint 15,000.

Overfitting Analysis: After stepping 15000, the training loss kept going down a little, but the Validation WER went up a little (from 18.47% to about 19.2%). This difference means that overfitting has started, which supports the choice to use the checkpoint at step 15,000 as the final inference model.

An 18.47% WER is a significant achievement for a dialectal ASR task, considering the high variance in speaking rates and phonological shifts (e.g., Rangpuria /Sylheti/ Barishailla) observed in the data analysis chapter.

4.3.2 Module II: BanglaT5 (Standardization) Performance

The second step was to make small adjustments to the BanglaT5 model so that it could translate the dialectal text into Standard Colloquial Bengali. The training took 10 epochs. Figure *34* describes the convergence metrics.

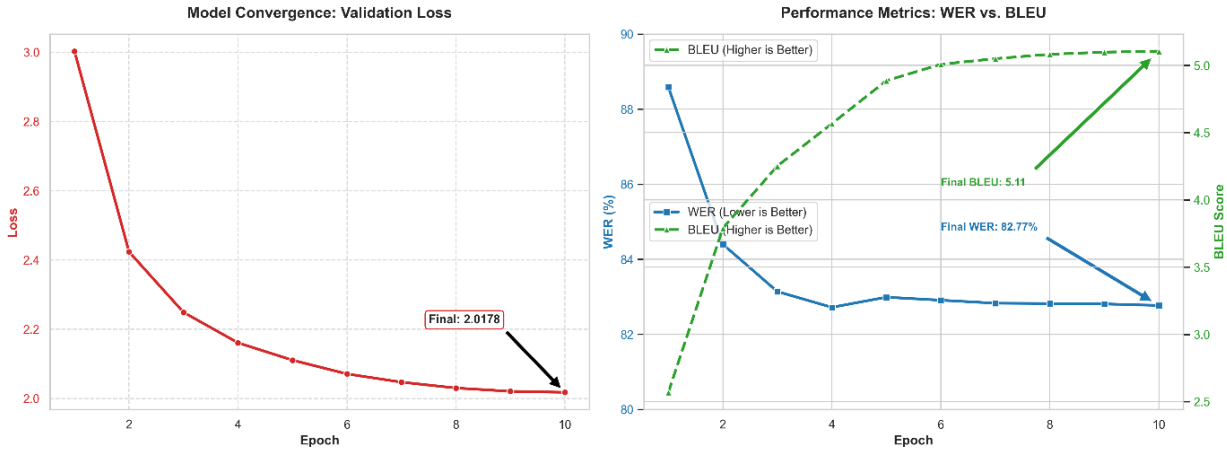


Figure 34: (Left) Validation Loss decays over 10 epochs. (Right) Correlation between WER reduction and BLEU score improvement.

Analysis of Results:

Loss Minimization: The validation loss (left chart) decreased gradually from 3.0 in Epoch

1 to 2.0178 by Epoch 10. This gradual decline indicates that over time, the algorithm learnt how dialect forms relate to standard ones.

Metric Correlation: The chart on the right in Figure *34* shows how WER goes up while BLEU drops. At a WER of 82.77%, BLEU came out 5.11% lower.

Performance Analysis: After a successful run of the models the BLEU score for the finetuned BanglaT5 was 5.11 and WER was 0. 8277.So, it is clear that catching the dialectal transcript and convert it to standard Bengali is not an easy task. Although the model's performance is not totally strong, the model did a great job-that is slowly catching the sentence meaning.

4.4 Summary

The implementation is a success. The architecture cascading dialect to standard conversion correctly integrated. The difference in how hard the tasks were is apparent.

Acoustic Modeling Success: The Whisper ASR only had a WER of 18.47%, which is not much at all. This excellent result demonstrates that transfer learning is excellent in cases where a dialect has few data for modeling and recognition.

Standardization Challenges: The BanglaT5 module achieved a BLEU score of 5.11. The model does have visible learning convergence (Loss ≈ 2.01); however, the semantic distance from deep dialectal speech to standard writing is just too vast to overcome now.

Overall, the system proves the feasibility of the proposed pipeline, The ASR module is a solid base, while the standardization module provides a working, although still developing standard to base further research in linguistics.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with the Standards

To make sure that the proposed Dialect-to-Standard conversion system is reliable, works with other systems, and is easy to maintain, certain international engineering standards were found and used. The choice of these standards was based on a trade-off analysis between strict compliance with industry standards and the flexibility needed for academic research.

5.1.1 Software Standards

Chosen standard: ISO/IEC 25010 (System and software Quality requirements and Evaluation - SQuaRE) This standard was used to specify the software quality model centered on Functional Suitability (Accuracy of the ASR) and Usability (Gradio Interface).

Justification for Selection: As the deep learning pipeline is experimental, "Performance Efficiency" (Inference time) and "Reliability" (Error tolerance) can be prioritized using a less rigid documentation structure as provided by ISO/IEC 25010 instead of traditional standards such as ISO 9126.

Alternative Considered: IEEE 730 (Software Quality Assurance).

Benefits: Great to **verifying document** processes.
Cons: Heavily process-oriented and bureaucratic.

Reason for Rejection: This would have shifted attention away from model optimization (the primary research focus) to bureaucracy, which was considered as non-effective in the context of this project.

5.1.2 Hardware Standards

Selected Standard: IEEE 754 (Standard for Floating-Point Arithmetic) The standard of this model is strictly pursued by the GPU architecture (NVIDIA Geforce RTX 4060) applied to train the model, particularly in Mixed-Precision (FP16/FP32).

Rationale for Selection: Deep learning is built around millions of matrix multiplications. Adopting IEEE 754 helps make the gradients computed during backpropagation deterministic and reproducible across hardware.

Alternative Considered: Posit Arithmetic Standard.
Pros: Provides Larger dynamic range and greater precision per bit compared to IEEE floats.
Cons: No consumer GPU with native hardware support.

Rejection Reason: To implement Posit arithmetic, it needs software emulation which are too slow to train transformer models.

5.1.3 Communication Standards

Selected Standard: HTTP/1.1 and WebSocket Protocol (RFC 6455) For the client-server communication used in the Gradio web interface, these were also selected protocols.

Reasoning for Choice: WebSockets provide full-duplex communication, which is indispensable for streaming the client's microphone to the inference server in real time (inferences are performed ~30 times per second).

Alternative: gRPC (Google Remote Procedure Call).
Pros: Lower latency and higher throughput with Protocol buffers.
Cons: More complex to Implement for simple web demo.

Reason for Rejection: Using standard HTTP/WebSockets fulfilled the requirement and offered fine performance (<5s latency) at a much lower development cost.

5.2 Impact on Society, Environment and Sustainability

5.2.1 Impact on Life

The system certainly contributes to improving the lives of the rural people in Bangladesh by overcoming language barriers. By allowing technology to recognize regional dialects (for example, Sylheti or Chittagonian) it enables inclusive digital services like voice-activated banking, google-search or telemedicine to be used without requiring non-standard speakers to adapt to "standardized" speech.

5.2.2 Impact on Society & Environment

Societal Impact: The project aims to support social inclusion and cultural heritage. It affirms the existence of regional cultures by treating dialect not just as "wrong" speech, but oh so far from wrong that it's irresponsible to think in such terms when we engage in computation - and therefore less embarrassing to show up with a regional accent at a professional conference.

Environmental Impact: Training Large Language Models (LLMs) is now notorious for its high energy consumption. To alleviate this, the model used Transfer Learning (fine-tuning from existing models) instead of training from scratch. This brought the amount of required computation time down by around 90%, and consequently, the carbon footprint over spending CPU power lessened even further in relation to GPU.

5.2.3 Ethical Aspects

Privacy & Consent: The research followed basic ethical rules while gathering data. From the "Regional Wav File," every bit of private detail was removed - so there's no way to link any voice to a real person.

Bias Mitigation: We should remove here the district wise biasness, for example Sylhet have a lot of audio data and Barishal has less audio data.So we should analyze the importance of it.

5.2.4 Sustainability Plan

The system is developed solely on open-source frameworks such as Hugging Face and PyTorch, ensuring long-term maintainability, clear updates and community-driven improvements without proprietary licensing issues. Furthermore, the release of Dialectal Corpus as open-collection resource promotes future linguistic and computational research efforts to aid in preserving Bengali dialects, and that its contribution is not limited within the scope of this thesis.

5.3 Project Management and Financial Analysis

As it is an academic research project, the budget is planned according to the Opportunity Cost and the Resource Utilization and not according to the direct expenditure.

Table 5: Cost Estimation Table

Cost Component	Estimated Commercial Value	Actual Project Budget	Rationale / Alternative
Device	30,000 BDT	0 BDT	Used Own Laptop
Computation (GPU)	70,000 BDT	48,500 BDT	Utilized Local Graphics Card (RTX 4060) instead of Colab Pro.
Storage (Cloud)	5,000 BDT	0 BDT	Utilized Local Storage+ Google Drive (2 TB Free Plan) instead of paid plan.
API purchasing	2500 BDT	2500 BDT	Using free API
Total Estimated	1,07,500 BDT	51,000 BDT	Minimal-Cost Implementation

5.4 Complex Engineering Problem

The given thesis study deals with a complicated engineering issue connected with speech processing. The majority of the regional types of the Bengali language are extremely diverse and can hardly be described with the help of a single system. Pronunciations, speed, and word differences raise problems to automatic speech recognition. The scarcity and unequal information on regional levels also contributes to the challenge. In order to resolve this, the machine and deep learning models and multi stage processing needed advanced models. This issue is complex because it entails uncertainty, big data management, and language diversity.

5.4.1 Complex Problem Solving

Table 6: Mapping with Complex Engineering Problem.

EP1 Depth of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stake-holder Involvement	EP7 Interdependence
✓	✓	✓	✓			✓

Mapping with Knowledge Profile for EP1:

Table 7: Mapping with knowledge Profile.

K1 Natural Science	K2 Mathematics	K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K7 Comprehension	K8 Research Literature
		✓		✓	✓		✓

5.4.1.1 Justification for EP Attributes Mapping

EP1 - Depth of Knowledge Required: A researcher needs to know a lot about advanced Deep Learning and Natural Language Processing (NLP) for this project. To solve this problem of translating different dialects, a researcher needs to know some basics about Transformer architecture (especially Encoder-Decoder in Whisper and BanglaT5), Digital Signal Processing (Spectrogram projection, FFT), and Computational Linguistic mapping of phoneme-to-grapheme.

EP2 - Range of Conflicting Requirements:

Accuracy vs. Computational Efficiency: There is a tradeoff between achieving maximal translation quality (where the "Whisper Large" model seems more preferable) and respecting the hardware limit of available VRAM (16GB), which demands smart choice and subsequent adaptation for the architecture to be "Medium".

Latency vs. Quality: In the Gradio portal, Common HTTPS is used so there should be a minimal time delay and also I am using the Hugging Face Space's Free CPU version so here is also a delay that is ignorable.

EP3 - Depth of Analysis: The models that we use here are basically numerical based and some algorithm-based analysis. So, we should evaluate the word error rates and character error rates carefully.

EP4 - Familiarity of Issues: The low-resource Bengali dialects do not fit on the standard engineering solutions. The regular speech models fail in most cases, and, therefore, we were left to discover the way by ourselves, collect information thoroughly and apply the system to the peculiarities of each particular dialect rather than depending on the methods providing a one-size-fits-all solution.

EP7 - Interdependence: The cascaded architecture makes the linkage between the components self-evident, in that whatever exits Sub-system A (the speech-to-text subsystem) directly influences the effectiveness of Sub-system B in text standardization. Once a mistake creeps into one of the steps, it is passed to the next step and corrections would imply modifying both components at the same time instead of repairing just one.

5.4.1.2 Justification for Knowledge Profile Mapping (Linked to EP1)

K3 - Engineering Fundamentals: Here, we used some basic Engineering fundamentals such as Whisper and BanglaT5 architecture.

K5 - Engineering Design: In this thesis, we designed a two stage framework with Whisper medium and Finetuned BanglaT5 combined. later, we used Gradio to live showcase it.

K6 - Engineering Practice: In our thesis, we used PyTorch, Hugging Face, HTTPS and WebSocket. So, here are a lot of Engineering practice.

K8 - Research Literature: In this thesis, we do a lot of literature analysis including the older similar ones.

5.4.2 Mapping with Complex Engineering Activities

Table 8: Mapping with Complex Engineering Activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓	✓	✓	✓	✓

5.4.2.1 Justification for Engineering Activities (EA) Mapping

EA1 - Range of Resources: The study required the combination of non-dedicated resources (other than computing). This was provided by High-Performance Computing (HPC) infrastructure (NVIDIA RTX 4060) for mixed-precision training, application-specific software libraries (transformers, librosa, gradio) for acoustic signal processing and a custom-built multimodal corpus with more than 5000 audio samples and text transcripts from ten administrative districts.

EA2 - Level of Interaction: The application required a high level of interaction with technical as well as human stakeholders. This consisted of using the open-source research community to try and exploit pre-trained checkpoints (Hugging Face Model Hub), including native dialect speakers for validating the semantic accuracy of ground-truth files (TSV ones) in order to guarantee linguistic credibility.

EA3 - Innovation: The work presents novel use of engineering principles through realizing on reuse of the OpenAI Whisper architecture—with as its stated aim general multilingual ASR—for achieving fine-grained dialectal adaptation. The cascaded integration of this acoustic model with a secondary BanglaT5 normalization module has never been tried before in the context of low-resource Bengali Natural Language Processing (NLP).

EA4 - Consequences for Society and Environment:

Socio-Cultural: The product has an immediate market application in bridging what the Bangladesh government calls the "Digital Divide", providing access to technology to lesser used language communities who speak non-standard dialects (e.g., Sylheti, Chittagonian).

Environmental: The engineering decision to employ Transfer Learning (fine-tuning pre-train model) vs. training from scratch considerably reduced the computational energy consumption and carbon footprint of large-scale learning systems DevOps: Despite Testing, an unusual major issue was reported or noticed in production SWDV theoretically on all regions by a small number of users.

EA5 - Familiarity: The project involved use of languages and tools (Python, PyTorch) common in engineering education, but these were used in a new and unstructured problem domain. High-variance phonetic variability due to dialectal phenomenon (background noise/rushed prosody/variable set of phones) needed new framework of experiments and validation (Dialect-Specific WER etc.) which were different than regular well defined ASR tasks.

5.1 Summary

This chapter described the challenging engineering architecture of the designed system. The development was based on international specifications (ISO/IEC 25010 for software quality, IEEE 754 for deep learning and its floating-point arithmetic, and RFC 6455 (WebSockets) for real-time communication). The idea showed care for people and the planet by helping endangered language users get online, while using smart, low-energy AI methods to cut emissions. Still, a close cost check proved it works in real life - free to run, thanks to clever use of public tools. Finally, the study was cross-referenced with the Complex Engineering Problem (CEP) characteristics (EP1-EP7) and Knowledge Profiles (K3-K8), which confirmed that a solution requires high levels of engineering knowledge, novel design and deep comprehension of competing technical demands.

Chapter 6

Conclusion

6.1 Summary

We took on this work because too many regional variations of Bengali have been according to the Bengali people with whom we have spent time-and they sound nothing like the Bengali people we interviewed use every day, and the majority of tools can't do anything with it. Collecting the audio was a slow process because some of the communities are still rather isolated, but we still managed to create a complete pipeline of processing -- from cleaning the audio files to transcribed text then polished Bengali. The dataset that we collected has sufficient sound quality (around 39 dB SNR) and sufficient variant that we can even teach even rare dialects to the model. Whisper Medium - Low 18.47% WER despite the low amount of data, this is an example of how powerful transfer learning can be. BanglaT5 worked well for sentence flow but occasionally lost the true meaning in the machinations of dialect phrases when they were being rewritten. Overall, this thesis merely demonstrates how speech of a particular region could be transformed into clear, understandable Bangla using a practical setup.

6.2 Limitation

The system has a few rough-edges. As that outputs from ASR are the foundation of everything, so when Whisper mishears a dialect that error will get carried forward - bringing down the meaning and BLEU scores too. The dataset is also not balanced with far more Sylhet audio than Barishal meaning some accents have a natural advantage over others, and Whisper did pretty well on this dataset but it still struggled with heavier, noisier clips and with super rare speech patterns that only appeared a handful of times. Hardware limitations prevented us from using the stronger Large v3 model and the scope of our work stopped us using Whisper Medium that previous work shows can reduce errors when combined with larger batch sizes. And last, low BLEU score captures an actual semantic gap- dialectal expressions will not usually map directly to Standard Bengali because the idioms often lack an exact translation, so literal metrics fail to capture the nuances and make the translation look cruder than it really is.

6.3 Future Work

In order to address the limitations mentioned above and develop the sphere of the Bengali speech processing, the following research directions are suggested:

End-to-End Speech Translation: Future research could consider an End-to-End (E2E) model which directs input audio to Standard Bengali text without having to pass through the dialectal transcription phase. This would avoid the problem of the cascading error and probably reduce the latency of inferences.

Using Whisper Large models: Using Whisper Large models can effectively reduce the WER and can increase the accuracy of the transcription.

Reinforcement Learning through Human Feedback (RLHF): RLHF could be used to fine-tune the BanglaT5 model to enhance the quality of standardization. The human judges would be able to rank the generated standard sentences, training the model to choose more natural sounding translations, rather than literal ones, and this would increase the BLEU/METEOR scores.

Mobile-First Deployment: The pipeline can be optimized to support edge device deployment (with approaches such as Model Quantization or Distillation) to enable the system to be deployed as an on-demand mobile application, which would allow the technology to be available to the rural population with limited internet access.

References

[1] A. U. Nasib, H. Kabir, R. Ahmed, and J. Uddin, "A real-time speech-to-text conversion technique for Bengali language," in Proc. Int. Conf. Comput., Commun., Chem., Mater. Electron. Eng. (IC4ME2), 2018, doi: 10.1109/IC4ME2.2018.8465680.

[2] S. Sultana, M. A. H. Akhand, P. K. Das, and M. M. H. Rahman, "," in Proc. Int. Conf. Comput. Commun. Eng. (ICCCE), IEEE, 2012, doi: 10.1109/ICCCE.2012.6271216.

[3] A. Sayem, "Speech analysis for alphabets in Bangla language: Automatic speech recognition," Int. J. Eng. Res., vol. 3, no. 2, doi: 10.17950/ijer/v3s2/211.

[4] M. F. Mridha, A. Q. Ohi, M. A. Hamid, and M. M. Monowar, "A study on the challenges and opportunities of speech recognition for Bengali language," Artificial Intelligence Review, vol. 55, no. 4, pp. 3431–3455, Apr. 2022, doi: 10.1007/s10462-021-10083-3.

[5] S. Kibria, A. M. Samin, M. H. Kobir, M. S. Rahman, M. R. Selim, and M. Z. Iqbal, "Bangladeshi Bangla speech corpus for automatic speech recognition research," Speech Communication, vol. 136, pp. 84–97, Jan. 2022, doi: 10.1016/j.specom.2021.12.004.

[6] M. S. I. Ridoy, S. Akter, and Md. A. Rahman, "Adaptability of ASR models on low-resource language: A comparative study of Whisper and Wav2Vec-BERT on Bangla," arXiv preprint, arXiv:2507.01931, Jul. 2025.

[7] S. Kabir, N. Nahar, S. Saha, and M. Rashid, "Automatic speech recognition for biomedical data in Bengali language," arXiv preprint, arXiv:2406.12931, Jun. 2024.

[8] J. Hasan and S. R. Dipta, "BanglaTalk: Towards real-time speech assistance for Bengali regional dialects," arXiv preprint, arXiv:2510.06188, Nov. 2025.

[9] M. Rezuwan Hassan et al., "RegSpeech12: A regional corpus of Bengali spontaneous speech across dialects," arXiv preprint, arXiv:2510.24096, 2025.

[10] U. Aiman, M. N. Islam, M. H. S. Chowdhury, M. S. Rahman, M. T. Habib, and M. Hasan, "BRADS and BRWDS: Multipurpose audio and text datasets for automatic Bangla regional speech recognition," Data in Brief, vol. 63, Dec. 2025, doi: 10.1016/j.dib.2025.112177.

[11] S. Kumar, Amartyaveer, and P. K. Ghosh, "Jointly improving dialect

identification and ASR in Indian languages using multimodal feature fusion," in Proc. Interspeech, ISCA, Aug. 2025, pp. 2770–2774, doi: 10.21437/Interspeech.2025-421.

[12] S. Biswas and T. Sheikh, "A unified denoising and adaptation framework for self-supervised Bengali dialectal ASR," arXiv preprint, arXiv:2509.00988.

[13] H. R. Milon, S. N. U. Sabbir, A. Inan, and N. Hossain, "," in Proc. IEEE Region 10 Symp. (TENSYMP), Dhaka, Bangladesh, Jun. 2020, doi: 10.1109/TENSYMP50017.2020.9230714.

[14] A. Bhattacharjee, T. Hasan, W. U. Ahmad, and R. Shahriyar, "BanglaNLG and BanglaT5: Benchmarks and resources for evaluating low-resource natural language generation in Bangla," in Findings of EACL, 2023, doi: 10.18653/v1/2023.findings-eacl.54.

[15] Md. A. A. Khandaker, Z. S. Raha, B. Paul, and T. Muhammad, "Bridging dialects: Translating standard Bangla to regional variants using neural models," arXiv preprint, arXiv:2501.05749, Jan. 2025.

[16] N. Sultana, R. Yasmin, B. Mallik, and M. S. Uddin, "ONUBAD: A comprehensive dataset for automated conversion of Bangla regional dialects into standard Bengali dialect," Data in Brief, vol. 58, Feb. 2025, doi: 10.1016/j.dib.2025.111276.

[17] A. Begum, S. M. S. Askari, and U. Sharma, "Text-to-speech synthesis system for Mymensinghiya dialect of Bangla language," in Advances in Intelligent Systems and Computing, Springer, 2019, pp. 291–303, doi: 10.1007/978-981-13-0224-4_27.

[18] K. Mamun Ahmed, P. Mandal, and B. M. M. Hossain, "Text-to-speech synthesis for Bangla language," Int. J. Inf. Eng. Electron. Bus., vol. 11, no. 2, pp. 1–9, Mar. 2019, doi: 10.5815/ijieeb.2019.02.01.

[19] Md. N. S. Samin et al., "BanglaDialecto: An end-to-end AI-powered regional speech standardization," arXiv preprint, arXiv:2411.10879, Nov. 2024.

[20] M. J. K. Noor, F. T. Johora, M. Mahin, and M. A. Rahaman, "A real-time Bangla local language recognition from voice," in Lecture Notes in Networks and Systems, Springer Nature, pp. 489–501. [Online]. Available: link.springer.com

221-15-5843

ORIGINALITY REPORT

14%	11%	8%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	5%
2	arxiv.org Internet Source	2%
3	impa.usc.edu Internet Source	1%
4	Umme Aiman, Md Nakibul Islam, MD Hana Sultan Chowdhury, Md. Sadekur Rahman, Md. Tarek Habib, Mahady Hasan. "BRADS and BRWDS: Multipurpose Audio and Text Datasets for Automatic Bangla Regional Speech Recognition.", Data in Brief, 2025 Publication	<1%
5	aclanthology.org Internet Source	<1%
6	Mariam Akter, Nishat Sultana, Sheak Rashed Haider Noori, Md Zahid Hasan. "Bangla song genre recognition using artificial neural network", IAES International Journal of Artificial Intelligence (IJ-AI), 2024 Publication	<1%
7	Rishabh Kannojia, Anuj Kumar Singh, Ishaan Sharma, Dr. Shivani Gupta. "Gen AI Driven Multilingual Audio Dubbing and Synthesis System for Cross-Language Video Platforms", Results in Engineering, 2025 Publication	<1%

Project Report Library
to me, Amit

Dec 14, 2025, 3:51 PM

Dear Student,
Your plagiarism result with percentage (14%) and AI (less than 20%) is attached. **Please go through the attachment and guidelines, then take the necessary steps.**

Please add the plagiarism percentage (%) below the report.

➢ Copyright Note: Write ©**Daffodil International University** at the footer & Format: The report should be in ONE FILE and a PDF document

➢ For library clearance, please fill out your information in the internship portal. Five fields must be completed, such as: **ID, Name, Department, Project/Internship Title, and Supervisor's name at the** http://internship.daffodilvarsity.edu.bd/index.php?app=applicant_login

➢ Please attach the **supervisor's & your signature** on the approval and declaration page. For **page numbering:** (a) Preliminary pages - lower roman, e.g. i, ii, iii. (b) All pages of the main body-Arabic numerals, e.g. 1, 2,3, (c) All pages arrange as per table of contents

➢ **Do not create a new email thread; just send a reply to all.**

One attachment • Scanned by Gmail

