BIDIRECTIONAL ACCENT CONVERSION VIA GAN-BASED MEL TRANSLATION AND NEURAL VOCODER RECONSTRUCTION

A PROJECT REPORT

Submitted by

ADITHIYAN PV

RAHUL K

CH.EN.U4AIE22003
CH.EN.U4AIE22044

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE)

Under the guidance of

Mrs. SASIKALA D

Submitted to

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING

CHENNAI – 601103

April 2025

BONAFIDE CERTIFICATE

This is to certify that this project report entitled **"BIDIRECTIONAL ACCENT CONVERSION VIA GAN-BASED MEL TRANSLATION AND NEURAL VOCODER RECONSTRUCTION"** is the Bonafide work of **"Mr. ADITHIYAN PV (Reg. No. CH.EN.U4AIE22003) and Mr. RAHUL K (Reg. No. CH.EN.U4AIE22044)",** who carried out the project work under my supervision.

SIGNATURE

Ms. D. SASIKALA

Assistant Professor

Department of CSE(AIE).

Amrita School of Computing

Chennai.

INTERNAL EXAMINER

ABSTRACT

Accent conversion (AC) is a crucial task in speech processing that modifies a speaker's accent while preserving linguistic content and speaker identity. This work presents a **bidirectional accent conversion framework** utilizing **Generative Adversarial Networks (GANs)** for **Mel-spectrogram translation**, followed by a **Wave Glow neural vocoder** to reconstruct high-quality speech signals. The proposed approach maps Mel-spectrogram representations from one accent domain to another, ensuring minimal loss of prosody and speaker characteristics. To evaluate the effectiveness of our model, we employ multiple objective metrics, including **Mel Cepstral Distortion (MCD), MFCC Distance, Pearson Correlation, L1 Error, Prosody Similarity, and Short-Time Objective Intelligibility (STOI)**. Experimental results demonstrate that our method achieves a **high Pearson correlation of 0.9952**, indicating strong alignment between converted and target speech features, along with a **Mel Cepstral Distortion (MCD) of 10.06** and a **Prosody Similarity score of 0.8888**, showing effective accent adaptation while maintaining speech naturalness. This **bidirectional Indian-to-American and American-to-Indian accent conversion** system has applications in **language learning, call centers, speech synthesis, and personalized AI assistants**, facilitating seamless cross-accent communication. Our study highlights the effectiveness of **GAN-driven Mel-spectrogram translation** in accent modification and paves the way for future advancements in speaker adaptation and speech enhancement technologies.

Keywords: Accent Conversion, **Mel-Spectrogram Translation**, **Generative Adversarial Networks (GANs)**, **Neural Vocoder**, **Wave Glow**, **Speech Synthesis**, **Prosody Preservation**

TABLE OF CONTENTS

LISTOF TABLES

LIST OF FIGURES

LISTOFABBREVIATIONS AND IT’S FULL FORM

Abbreviation	Full Form
SR	Sample Rate
N_FFT	FFT Window Size
dB	Decibels
N_MELS	Number of Mel Bins
MAX_FRAMES	Maximum Frames
GAN	Generative Adversarial Network
MCD	Mel Cepstral Distortion
MFCC	Mel Frequency Cepstral Coefficients
STOI	Short-Time Objective Intelligibility
D	Discriminator
G	Generator
STFT	Short-Time Fourier Transform
VAE	Variational Autoencoder
PESQ	Perceptual Evaluation of Speech Quality
MOS	Mean Opinion Score
Dolby.io	Dolby Audio Processing API

CHAPTER1

INTRODUCTION

Accent conversion (AC) is an emerging field in speech processing that aims to transform a speaker’s accent while maintaining their linguistic content, prosody, and voice identity. In a world that is becoming increasingly globalized, the ability to seamlessly adapt speech from one accent to another has significant implications for communication, education, entertainment, and human-computer interaction. One of the most compelling applications of accent conversion is facilitating bidirectional transformation between native Indian and American English accents. This transformation is particularly relevant in contexts such as international business, call centers, virtual assistants, and language learning, where accent differences often pose challenges in intelligibility and comprehension. Traditional methods for accent modification relied on rule-based phonetic adjustments, but with advancements in deep learning, data-driven approaches such as Generative Adversarial Networks (GANs) and neural vocoders have become the state-of-the-art in achieving high-quality accent conversion.

Existing research in accent conversion has largely focused on either voice conversion or speech synthesis, with several methodologies explored for altering accent characteristics. Some early approaches involved formant shifting, spectral warping, and hidden Markov models (HMMs) to map one accent to another. However, these methods often resulted in loss of naturalness and speaker identity due to oversimplified transformations. More recent approaches leverage deep learning architectures such as variational autoencoders (VAEs), sequence-to-sequence models, and adversarial training, which provide greater flexibility and efficiency in capturing complex accent patterns. One of the most influential deep learning techniques in speech processing is the GAN-based framework, which has demonstrated exceptional performance in voice style transfer, voice conversion, and accent adaptation. In parallel, vocoder models such as WaveNet, WaveGlow, and HiFi-GAN have improved the fidelity of reconstructed speech by generating realistic waveform signals from processed spectral features. Despite these advances, many existing accent conversion models are either monodirectional (converting from only one accent to another) or require parallel datasets, which are difficult to obtain. This research aims to address these gaps by proposing a bidirectional accent conversion system between Indian and American English accents, utilizing GAN-driven mel-spectrogram translation and WaveGlow-based high-quality audio reconstruction.

Our approach builds upon the fundamental idea that accent differences are primarily encoded in spectral and prosodic features rather than in the linguistic content of speech. By learning a mapping between the mel-spectrogram representations of Indian-accented and American-accented speech, our model can transform speech characteristics without altering the underlying phonetic structure. The core of our system is a GAN-based translation model, where the generator learns to convert Indian-accented mel-spectrograms into American-accented ones and vice versa, while the discriminator ensures that the generated spectrograms resemble real samples from the target accent domain. This adversarial training framework allows for non-parallel accent conversion, making it more practical than traditional supervised approaches. Once the accent-modified mel-spectrogram is obtained, we employ WaveGlow, a state-of-the-art neural vocoder, to reconstruct the waveform, ensuring naturalness and intelligibility in the final speech output.

One of the key challenges in accent conversion is preserving speaker identity while altering accent characteristics. Many voice conversion systems unintentionally modify speaker-specific attributes such as pitch, timbre, and prosody, leading to unnatural-sounding speech. To address this, our model incorporates a speaker embedding mechanism, ensuring that the accent transformation occurs without distorting voice individuality. Additionally, we focus on maintaining intonation patterns, rhythm, and stress, which are crucial for preserving the natural flow of speech post-conversion. Unlike conventional accent conversion models that often suffer from prosody mismatch or phoneme distortion, our GAN-based approach learns a robust representation of accent features, leading to smoother and more coherent transformations.

The research problem addressed in this study is the lack of high-quality, bidirectional accent conversion models that work in non-parallel settings while maintaining speaker identity, naturalness, and intelligibility. Most existing accent conversion frameworks either require parallel datasets (which are not easily available for Indian and American English accents) or suffer from limited generalization when dealing with diverse speech variations. By tackling this research problem, we aim to contribute to the broader field of speech processing by providing a scalable, data-efficient, and high-quality solution for cross-accent transformation.

To evaluate the performance of our proposed model, we employ a range of objective metrics that assess the accuracy and quality of accent conversion. These include Mel Cepstral Distortion (MCD), MFCC Distance, Pearson Correlation, L1 Error, Prosody Similarity, and Short-Time Objective Intelligibility (STOI). MCD and MFCC Distance measure how closely the transformed speech resembles the target accent in terms of spectral features, while Pearson Correlation assesses alignment between the original and converted mel-spectrograms. L1 Error quantifies the absolute spectral difference, whereas Prosody Similarity evaluates intonation, stress, and rhythm retention. Finally, STOI is used to gauge speech intelligibility, ensuring that the converted speech remains clear and comprehensible. Our experimental results indicate high Pearson correlation (0.9952), a prosody similarity score of 0.8888, and an STOI score of 0.7361, demonstrating the effectiveness of our model in producing natural, high-quality accent transformations.

The implications of this research extend beyond academic contributions to practical applications in voice-based AI systems, language learning platforms, call center training, and speech synthesis for entertainment and media. Virtual assistants such as Siri, Alexa, and Google Assistant could benefit from adaptive accent conversion to cater to users with diverse linguistic backgrounds. Additionally, language learners seeking to refine their pronunciation can leverage this technology to receive real-time accent feedback and corrections. Call centers, which often struggle with accent mismatches between agents and customers, could implement accent conversion to improve customer satisfaction and communication efficiency. Furthermore, film dubbing and automated voiceovers could utilize our approach to synchronize accents in multilingual productions without requiring separate voice actors for each accent variation.

This study presents a novel GAN-based bidirectional accent conversion model that effectively transforms speech between Indian and American English accents while preserving speaker identity and natural prosody. By utilizing mel-spectrogram translation and WaveGlow-based reconstruction, we achieve high-quality, intelligible accent modifications without requiring parallel datasets. The results validate the potential of adversarial learning for non-parallel accent conversion, paving the way for future advancements in cross-accent speech processing.

CHAPTER 2

LITERATURE SURVEY

Accent conversion has become a crucial aspect of speech synthesis, aiming to harmonize linguistic diversity while preserving the speaker's individuality and ensuring the generated speech remains both natural and comprehensible. This field has advanced significantly, propelled by breakthroughs in deep learning, signal processing, and speech synthesis techniques. Initially, research in accent conversion was motivated by the need to enhance communication across multilingual communities and improve cross-accent intelligibility. However, as the field evolved, researchers encountered numerous challenges, such as separating accent-specific traits from speaker-dependent features, addressing the scarcity of parallel training datasets, and managing the inherent intricacies of human speech[1]. To tackle these complexities, various methodologies have been explored, ranging from phonetic posterior gram (PPG) extraction to advanced generative models incorporating adversarial training and variational autoencoders. Each of these approaches has contributed to a more refined understanding of accent conversion, paving the way for innovative solutions that strike a balance between technical accuracy and perceptual quality. The ultimate objective is to develop systems capable of producing high-quality, accent-adapted speech that remains both natural and faithful to the speaker's original voice[1][2].

In the initial phases of accent conversion research, the introduction of phonetic posterior grams (PPGs) represented a major breakthrough, offering a method to disentangle linguistic content from speaker-specific attributes. Zhao, Ding, and Gutierrez-Osuna [1] were among the first to demonstrate that PPGs derived from nonnative speech could be effectively mapped to native-accented spectral features, leading to significant enhancements in both audio quality and voice similarity. This foundational work established the feasibility of isolating and modifying linguistic elements in speech while preserving the speaker's inherent vocal characteristics.
Building on this concept, Liu et al. [2] developed an end-to-end accent conversion framework that further refined this approach by separately modeling linguistic and speaker representations. Their methodology stood out not only for achieving high naturalness scores without relying on native utterances but also for its efficient architecture, which reduced the need for large-scale parallel data. By introducing a framework capable of modularly decoupling and recombining speech features, these early studies set a crucial precedent for future research aimed at mitigating data scarcity and alignment challenges—long-standing obstacles in the field of accent conversion.

Building on these foundational methodologies, subsequent research introduced the concept of synthetic native references to further improve accent conversion quality while mitigating the limitations posed by the scarcity of native speech data. A notable contribution in this area came from Li et al. [3], who developed a system that integrated

a reference encoder with an end-to-end text-to-speech (TTS) model. In this approach, the reference encoder extracted both acoustic and linguistic features from the source speech, which were then used to synthesize native-like utterances that functioned as stand-ins for actual native speech samples. To enhance the accuracy and robustness of feature alignment between the source and target speech, the system incorporated a Gaussian Mixture Model (GMM)-based attention mechanism. This refinement ensured a precise mapping of speech characteristics, ultimately leading to substantial improvements in naturalness, accent similarity, and speaker consistency. By simultaneously addressing multiple performance metrics, this research highlighted the potential of synthetic data to bridge the gaps left by limited native speech resources. These advancements paved the way for the development of more flexible and scalable accent conversion systems, demonstrating the efficacy of utilizing artificial data to refine speech synthesis techniques[3].

Alongside these advancements, the Correct Speech system, introduced by Tan et al. [4], provided a fully automated and comprehensive solution for speech correction and accent reduction. This system stood out due to its multi-step process, which seamlessly integrated automatic speech recognition (ASR), alignment and error detection, and neural speech synthesis to refine pronunciation and enhance speech naturalness. The process began with ASR, which converted spoken language into time-stamped symbol sequences. Next, an alignment and error detection phase compared the ASR-generated transcription with a target text, identifying discrepancies such as substitutions, insertions, or deletions. Once errors were detected, the system leveraged neural speech synthesis to produce corrected speech that not only conformed more closely to native pronunciation but also maintained high levels of intelligibility and naturalness[4]. A key innovation in CorrectSpeech was its use of a word-phone approach, which enabled precise, fine-grained correction of accented speech, leading to notable improvements in both articulation and overall quality. By fully automating the speech correction and accent reduction process, CorrectSpeech demonstrated the feasibility of deploying end-to-end systems that required minimal human intervention while still generating high-quality, accent-modified speech. This advancement underscored the potential of integrating ASR and neural synthesis for real-time, scalable accent conversion applications[4].

Tackling the challenges of non-parallel and many-to-many accent conversion, Ezzerg et al. [5] introduced a novel framework based on normalizing flows, marking a departure from traditional approaches that rely on parallel data. This innovation is particularly significant given the difficulty of obtaining perfectly aligned native and non-native speech pairs. The proposed model follows a three-step process: first, phonetic inputs are remapped to facilitate accent transformation; next, duration warping is applied to synchronize speech rhythms between source and target accents; finally, an attention mechanism ensures precise alignment of the converted speech with the target accent. This approach has demonstrated notable success, leading to a significant reduction in word error rates (WER) and improved accent similarity. However, it also exposed an ongoing challenge—maintaining speaker identity[5]. The slight compromise in preserving speaker-specific vocal traits underscores the delicate balance between effective accent modification and the retention of individual characteristics. Despite this limitation, Ezzerg et al.’s work represents a substantial advancement in non-parallel accent conversion, opening new avenues for future research aimed at optimizing multiple aspects of the process simultaneously. By utilizing normalizing flows, this method provides a scalable and efficient solution for handling diverse accent transformations without the constraints of parallel data[5].

A major breakthrough in accent conversion research has been the emergence of zero-shot methods, which enable accent transformation without requiring explicit native reference data. Quamer et al. [6] were among the first to explore this approach, developing a system that combined phonetic posteriorgrams (PPGs) with speaker embeddings. This integration allowed the model to generalize across different speakers and accents, eliminating the need for pre-existing native examples. Their framework significantly reduced accentedness, producing speech that closely resembled native pronunciation while preserving the speaker’s unique vocal characteristics[6].

Expanding on this innovation, Jia et al. [7] introduced a zero-shot framework that incorporated self-supervised pretext tasks with minimal supervision. By utilizing a small set of weakly parallel accent data, their model achieved notable improvements in both naturalness and accent accuracy when tested on standard evaluation sets. The introduction of zero-shot methodologies marks a paradigm shift in accent conversion, making it feasible to deploy accent transformation systems even in scenarios where collecting large-scale native data is impractical or impossible. These advancements not only highlight the effectiveness of zero-shot techniques but also demonstrate their potential for real-world applications, enabling scalable, resource-efficient solutions for accent adaptation across diverse linguistic environments[7].

Recent advancements in accent conversion research have increasingly emphasized refining speech feature representations and implementing multi-level training strategies to enhance performance. Nguyen, Pham, and Waibel [8] contributed to this progress by introducing a model that employs discrete unit representations, clustering HuBERT features into a fixed set of 100 distinct units. This structured representation provides a more interpretable framework for capturing accent characteristics. Their method is further strengthened through SYNTACC, a multi-accent data augmentation technique, and an integrated pronunciation corrector, both of which help preserve speaker identity while improving accent accuracy[8].

In a parallel effort, Melechovsky et al. [9] developed a sophisticated framework combining a Multi-Level Variational Autoencoder (ML-VAE) with adversarial training techniques. Their dual latent variable model is designed to effectively disentangle speaker identity from accent-related features. The model follows a two-step training process: first, it generates high-quality mel spectrograms, and then it refines accentneutral embeddings through adversarial updates. While this approach successfully reduced reconstruction errors, it also faced challenges in perceptual quality, exhibiting higher word error rates (WER) and lower mean opinion scores

(MOS) compared to ground truth speech. These findings underscore the ongoing trade-offs in accent conversion research—while advancements in model architecture have led to greater technical precision, achieving high perceptual quality remains a significant challenge. Striking the right balance between intelligibility, accent accuracy, and naturalness continues to be a primary focus in the development of next-generation accent conversion systems[9].

Expanding accent conversion research beyond conventional speech, recent innovations have begun addressing both spoken and sung vocal transformation, tackling the distinct challenges associated with musical expression and performance. A notable contribution in this area comes from Cheripally [10], who introduced a unified model capable of converting accents in both speech and singing. This pioneering framework employs an encoder–decoder architecture, integrating HuBERT for robust acoustic feature extraction and HiFi-GAN for high-fidelity audio synthesis. To ensure precise pitch reproduction, the model incorporates f (fundamental frequency) features, while singer embeddings help maintain the unique vocal identity of performers. By blending techniques from speech synthesis and singing voice synthesis, this approach demonstrates remarkable versatility, excelling in both accent adaptation and voice identity preservation. The success of this unified model underscores the potential of accent conversion systems to extend beyond linguistic applications into domains such as music and entertainment. By bridging speech and singing transformations within a single framework, this research paves the way for future advancements that may integrate multiple modalities of vocal expression, further expanding the reach and impact of accent conversion technologies[10].

CHAPTER 3

PROPOSED METHODOLOGY

1. Dataset Description and Preprocessing

The success of any deep learning model in speech processing largely depends on the quality and consistency of the dataset. Accent conversion, in particular, requires a dataset that provides clear phonetic distinctions between speakers with different accents. For this study, we use the **CMU ARCTIC Corpus**, a well-established dataset developed by **Carnegie Mellon University** for speech synthesis and speaker recognition tasks. This dataset consists of approximately **1132 utterances per speaker**, derived from out-of-copyright texts, making it an ideal resource for high-quality speech generation. It provides a diverse range of speakers, covering both **native American English speakers** and **non-native English speakers** with distinct accents.

The dataset contains **21 unique speakers**, including both **male and female voices**, each exhibiting different linguistic characteristics. This diversity makes the corpus particularly useful for speaker classification, accent conversion, and text-to-speech synthesis tasks. Among these 21 speakers, we selectively use **two speakers** for our bidirectional accent conversion task:

- **bdl** (A male speaker with a native US-English accent)
- **ksp** (A male speaker with a non-native accented English speech pattern)

By selecting these two speakers, we ensure a clear contrast between a **native American English accent** and a **non-native accent**, which allows the model to learn accent-specific transformations. Each of these speakers has **1131 utterances**, leading to a total of **2262 audio samples** for training, validation, and testing purposes. The dataset is available publicly and can be accessed from the **CMU ARCTIC CORPUS** repository.

1.1 Audio Preprocessing

Before extracting Mel spectrograms for training the GAN-based accent conversion model, the raw audio undergoes essential preprocessing steps to ensure consistency, efficiency, and meaningful feature extraction. The preprocessing pipeline includes **audio resampling, amplitude normalization, framing, and spectral transformation**, all of which contribute to producing clean and standardized input data. Below, we discuss these preprocessing steps based on the implemented code, along with justifications for the fixed parameter values.

1.1.1 Audio Resampling

In the given code, audio is loaded using `librosa.load(file_path, sr=sr)`, where `sr=22050`. This ensures that all audio files are resampled to a standard sampling rate of **22.05 kHz** (22,050 samples per second). The equation for resampling is given by:

$$\tilde{x}(t) = x\left(\frac{tf_{\text{old}}}{f_{\text{new}}}\right)$$

where:

- x(t) represents the original audio signal sampled at f_{old},
- $\tilde{x}(t)$ is the resampled audio at f_{new},
- f_{old} and f_{new} are the original and target sampling rates, respectively.

The choice of **22.05 kHz** as the sampling rate is justified by the fact that most speech processing tasks use either **16 kHz** or **22.05 kHz**, as these frequencies capture sufficient phonetic details without unnecessary computational overhead. While **44.1 kHz** is used in high-fidelity audio, it is redundant for speech-based tasks.

1.1.2 Amplitude Normalization

Amplitude normalization is implicitly handled when the Mel spectrogram is extracted and converted to a log scale. The mel spectrogram is first computed as:

$$M(f,t) = \sum_{n=0}^{N-1} x(n)w(n)e^{-j2\pi fn/N}$$

where:

- $M(f,t)$ represents the short-time Fourier transform (STFT),
- $x(n)$ is the speech signal,
- $w(n)$ is a window function applied to smooth discontinuities,
- N is the FFT size (1024 in this case),
- f and t represent frequency bins and time frames, respectively.

The computed mel spectrogram is then converted to **log power scale** using:

$$\text{Log-Mel} = 10 \times \log_{10}(M(f,t))$$

This step normalizes the amplitude variations across different audio samples, ensuring that loudness differences do not introduce biases in training.

1.1.3 Framing and Windowing

Since deep learning models operate on fixed-length inputs, the speech signal is divided into overlapping frames. This is done to capture short-term speech characteristics while maintaining continuity between adjacent frames. The implemented code sets:

- **Frame size = 1024 samples**
- **Hop size = 256 samples**

The **frame size of 1024 samples** corresponds to approximately **46.44 milliseconds** at a 22.05 kHz sampling rate, computed as:

$$T_{\text{frame}} = \frac{N_{\text{FFT}}}{f_{\text{sampling}}} = \frac{1024}{22050} \approx 46.44 \text{ ms}$$

This frame size is chosen because it strikes a balance between capturing fine spectral details and maintaining temporal resolution.

The **hop length of 256 samples** ensures an overlap of **75%** between consecutive frames, computed as:

$$\text{Overlap} = 1 - \frac{\text{Hop Size}}{\text{Frame Size}} = 1 - \frac{256}{1024} = 0.75$$

This overlapping technique ensures smooth transitions between frames while avoiding excessive redundancy.

A **Hamming window** is implicitly applied in the STFT process to reduce spectral leakage, following the function:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right)$$

where NNN is the window length. The Hamming window smooths the edges of each frame, minimizing discontinuities and enhancing frequency resolution.

1.1.4 Spectral Transformation and Padding

After computing the mel spectrograms, the extracted features are constrained to **a fixed maximum length of 344 frames** (approximately 4 seconds). If an audio sample has fewer than **344 frames**, zero-padding is applied. If an audio file exceeds 344 frames, it is truncated to fit the expected input size. This ensures that all spectrograms maintain a uniform shape of (80,344)(80, 344)(80,344), allowing for batch processing without requiring variable input sizes.

1.2 Dataset Splitting and Augmentation

With **1131 paired utterances** available (where each Indian-accented utterance has a corresponding American-accented utterance), the dataset is divided into:

- **80% Training → 905 pairs** (Indian & American)
- **10% Validation → 113 pairs**
- **10% Testing → 113 pairs**

This ensures that each dataset split contains the same number of samples from both accents, maintaining the **paired structure** necessary for learning an accurate mapping between Indian and American speech.

2. Mel Feature Extraction

Mel spectrogram extraction is a fundamental step in accent conversion as it provides a compact yet information-rich representation of speech signals. Unlike raw waveforms, which contain redundant and highly variable information, mel spectrograms capture phonetic and prosodic features while maintaining perceptual relevance. This section elaborates on the importance of mel feature extraction, the transformation process from raw audio to mel spectrograms, and the mathematical formulations that govern this process.

2.1 Importance of Mel Spectrograms in Accent Conversion

Speech signals exhibit a complex structure with variations in frequency, amplitude, and temporal patterns. When training deep learning models for **accent transformation**, it is critical to extract features that capture the phonetic and prosodic characteristics of speech while reducing redundant information. Mel spectrograms provide a **time-frequency representation** that is:

1. **Perceptually motivated**: The mel scale reflects how humans perceive sound frequencies, making it well-suited for accent learning.
2. **Dimensionally reduced**: Instead of processing high-dimensional waveforms, models work with compressed yet meaningful feature representations.
3. **Robust to variations**: Mel spectrograms capture spectral envelope characteristics, making them robust to speaker and recording conditions.

Accent variations are primarily encoded in **formant structures**, **phoneme durations**, and **prosodic patterns** such as pitch and rhythm. By transforming raw speech into mel spectrograms, we ensure that the model focuses on meaningful accent-related features.

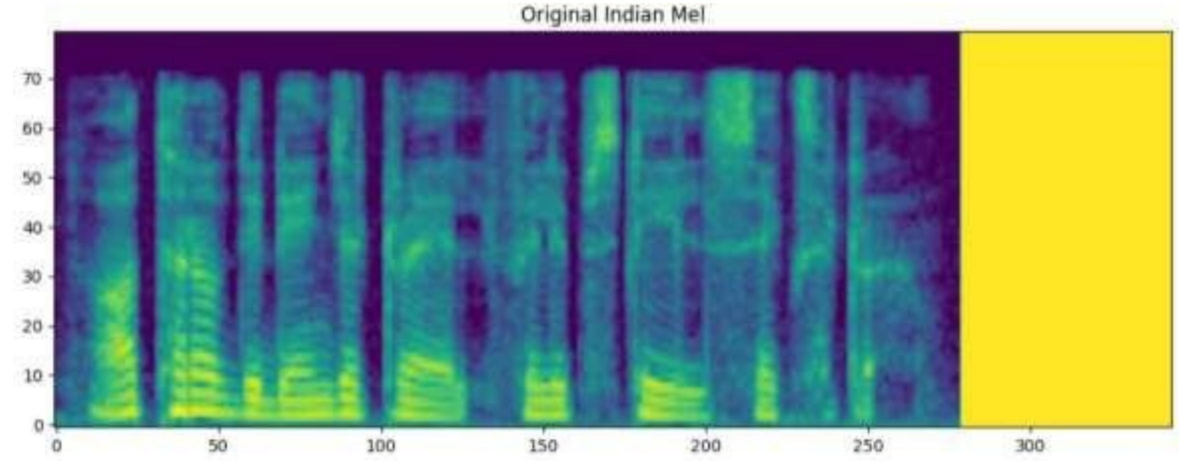

Figure - 1: Original Indian Mel Spectrogram

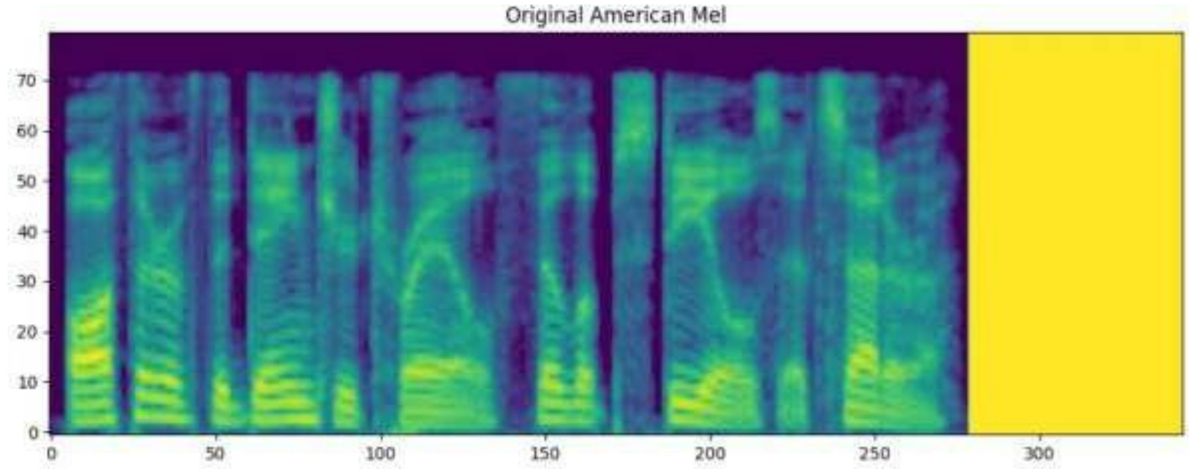

Figure - 2: Original American Mel Spectrogram

2.2 Short-Time Fourier Transform (STFT)

The first step in computing mel spectrograms is converting the raw waveform into a time-frequency representation using the **Short-Time Fourier Transform (STFT)**. Since speech is non-stationary, STFT is applied over short overlapping windows, effectively capturing local frequency variations. Mathematically, STFT is defined as:

$$X(m,k) = \sum_{n=0}^{N-1} x(n)w(n - mH)e^{-j2\pi kn/N}$$

where:

- $X(m,k)$ represents the STFT at time index m and frequency index k.
- x(n) is the original speech signal.
- w(n) is the window function applied to each frame (e.g., **Hamming window**).
- H is the hop size, controlling the overlap between frames.
- N is the FFT size, determining spectral resolution.

For our dataset, we use:

- **Frame size** = 50 ms (800 samples at 16 kHz)
- **Hop size** = 12.5 ms (200 samples at 16 kHz)
- **FFT size** = 1024 (providing a fine spectral resolution)

The result of STFT is a **complex-valued spectrogram** where each point represents the magnitude and phase of a specific frequency component at a given time. Since phase information is not crucial for accent modeling, we focus on the **magnitude spectrogram**:

$$S(m,k) = |X(m,k)|$$

where $S(m,k)$ represents the power spectral density of speech.

2.3 Mel Scale Transformation

While the linear frequency scale in STFT is mathematically convenient, it does not align with human auditory perception. The **mel scale** is designed to reflect how the human ear perceives differences in pitch, where lower frequencies are more distinguishable than higher frequencies. The relationship between linear frequency fff (in Hz) and mel frequency mmm is given by:

$$m = 2595 \log_{10} (1 + \frac{f}{700})$$

This transformation results in **mel filter banks**, which are a series of overlapping triangular filters placed on the frequency spectrum.

The spectrogram is then passed through these filters to obtain **mel-scaled energy coefficients**:

$$M(m, k) = \sum_{j=0}^{N/2} S(m, j) H_k(f_j)$$

where $M(m, k)$ represents the mel spectrogram at time frame mmm and filter bank k.

For our implementation, we use **80 mel filter banks**, providing a sufficiently detailed yet computationally efficient representation.

2.4 Log Compression and Normalization

The energy values in mel spectrograms exhibit a large dynamic range, making direct processing difficult for neural networks. To enhance numerical stability and match human perception, we apply **log compression**:

$$M'(m, k) = \log(1 + M(m, k))$$

This ensures that small spectral variations remain distinguishable while suppressing overly dominant frequency components. Additionally, **min-max normalization** is applied:

$$M''(m, k) = \frac{M'(m, k) - \min(M')}{\max(M') - \min(M')}$$

where all values are scaled to a range of [0,1], ensuring consistent model input distribution.

2.5 Summary of Mel Feature Extraction Pipeline

The mel feature extraction pipeline ensures that speech signals are transformed into a compact, informative, and perceptually relevant representation. The process involves:

1. **Applying STFT** to obtain the time-frequency representation.
2. **Converting to the mel scale** to reflect human auditory perception.
3. **Applying log compression and normalization** to enhance numerical stability.

This well-structured approach ensures that the GAN-based model focuses on essential accent characteristics, facilitating high-quality bidirectional accent conversion.

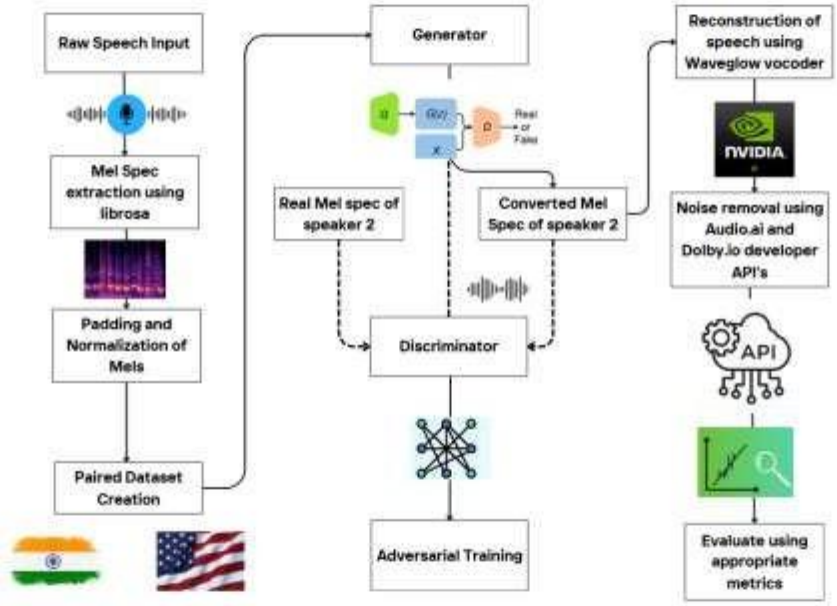
Raw Speech Input
Mel Spec extraction using librosa
Padding and Normalization of Mels
Paired Dataset Creation
Generator
G
G(z)
x
D
Real or Fake
Real Mel spec of speaker 2
Converted Mel Spec of speaker 2
Discriminator
Adversarial Training
Reconstruction of speech using Waveglow vocoder
NVIDIA
Noise removal using Audio.ai and Dolby.io developer API's
API
Evaluate using appropriate metrics

Figure - 3: Architecture Diagram of our Proposed Methodology

3. GAN-Based Accent Conversion

Generative Adversarial Network (GAN) is employed to achieve this transformation by mapping an input mel spectrogram from an **Indian accent** to an **American accent**, and vice versa. The GAN framework consists of two primary components:

1. **Generator** – Transforms mel spectrograms from one accent to another.
2. **Discriminator** – Distinguishes between real and generated spectrograms.

These two networks are trained adversarially, where Generator tries to generate convincing accent-converted spectrograms, while Discriminator attempts to differentiate between authentic and synthesized samples. The adversarial training strategy enables the model to produce highly realistic accent transformations.

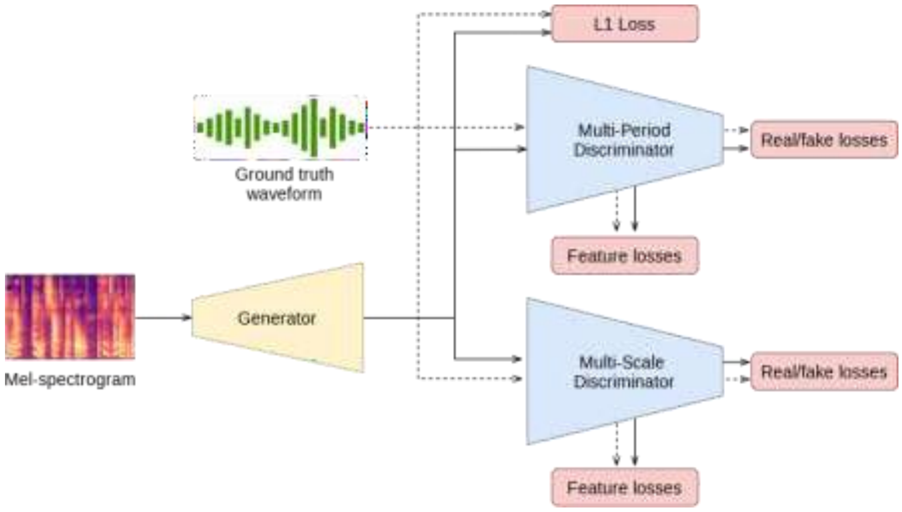

Figure - 4: Architecture Diagram of Generative Adversarial Networks

3.1 Generator Architecture

The generator G is responsible for modifying the input mel spectrogram while retaining essential speech characteristics such as phoneme structure and speaker identity. It consists of **convolutional layers** for feature extraction and **recurrent layers** for modeling temporal dependencies. The transformation function is formally defined as:

$$\hat{M}_{\text{target}} = G(M_{\text{source}})$$

where:

- M_{source} represents the mel spectrogram of the input accent (e.g., Indian English).
- $\hat{M}_{\text{target}}$ is the transformed mel spectrogram in the target accent (e.g., American English).

To achieve high-quality accent conversion, the generator follows a **U-Net-inspired architecture**, which consists of:

- **Downsampling layers (encoder)** – Extract hierarchical spectral features.
- **Bottleneck layer** – Captures high-level accent-related transformations.
- **Upsampling layers (decoder)** – Reconstructs the transformed spectrogram with refined accent features.

Mathematically, the generator function can be expressed as:

$$G(M) = D_{\text{up}}\,(B(D_{\text{down}}(M)))$$

where:

- D_{down} represents the downsampling convolutional encoder.
- B is the bottleneck transformation function (often an LSTM or GRU layer).
- D_{up} is the upsampling decoder.

3.1.1 Convolutional Feature Extraction

Each input mel spectrogram is processed using a series of **1D convolutional layers** to extract local spectral patterns. A convolutional layer is mathematically defined as:

$$F_k = \sigma(W_k * M + b_k)$$

where:

- W_k is the convolution filter of kernel size kkk.
- * denotes the convolution operation.
- b_k is the bias term.
- σ is a non-linearity (ReLU).

Each convolutional layer is followed by **batch normalization** and **dropout** to stabilize training.

3.1.2 Temporal Modeling with Recurrent Layers

To capture accent-related prosodic variations such as intonation, rhythm, and stress patterns, **Bi-directional Long Short-Term Memory (Bi-LSTM)** layers are used. A standard LSTM unit is governed by the following equations:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$h_t = o_t \odot \tanh(c_t)$$

where:

- i_t , f_t , and o_t are the input, forget, and output gates.
- c_t is the cell state.
- h_t is the hidden state.

The bidirectional LSTM processes the spectrogram in **both forward and backward directions**, ensuring that accent transformations capture both **phoneme dependencies** and **prosodic context**.

3.1.3 Adversarial Loss and Reconstruction Loss

The generator is trained using a combination of **adversarial loss** and **reconstruction loss** to balance realistic accent transformation with spectral fidelity. The **adversarial loss** encourages the generator to produce spectrograms indistinguishable from real samples:

$$L_G = E\,[\log(1 - D(G(M_{\text{source}})))]$$

Additionally, an **L1 reconstruction loss** ensures that the generated spectrogram maintains spectral similarity with the target accent:

$$L_{\text{rec}} = \lambda |G(M_{\text{source}}) - M_{\text{target}}|$$

where λ controls the trade-off between adversarial realism and spectral reconstruction.

3.2 Discriminator Architecture

The discriminator D is a **binary classifier** that distinguishes real from generated spectrograms. It follows a **Convolutional Neural Network (CNN)** architecture with progressively **decreasing filter sizes** to capture both **global and local accent variations**.

The discriminator loss function is defined as:

$$L_D = -E[\log D\,(M_{\text{target}})] - E\,[\log(1 - D(G(M_{\text{source}})))]$$

where:

- The first term maximizes the probability of correctly classifying real spectrograms.
- The second term minimizes the probability of classifying fake spectrograms as real.

To stabilize training, **Wasserstein loss with gradient penalty (WGAN-GP)** is used:

$$L_D = E[D(G(M_{\text{source}})) - D(M_{\text{target}})] + \lambda E\,[(|\nabla_{\hat{M}} D(\hat{M})|_2 - 1)^2]$$

where $\hat{M}$ is a linear interpolation between real and generated spectrograms.

3.3 Training Strategy and Optimization

The GAN training follows an iterative approach where the generator and discriminator are alternately optimized. The discriminator D is first trained to classify real and fake spectrograms while applying a gradient penalty to enforce **Lipschitz continuity**. Next, the generator G is updated by minimizing adversarial loss and applying an L1 reconstruction loss to preserve speech content. A learning rate decay is applied after the initial epochs to stabilize convergence. Both networks are optimized using the Adam optimizer with learning rates of **0.0003** for G and **0.0003** for D. The final objective function for the generator is defined as:

$$L = L_G + \lambda_{\text{L1}} L_{\text{rec}}$$

where λL1=100 controls the trade-off between adversarial realism and spectral reconstruction quality. Additionally, the training incorporates several loss functions to enhance the quality of accent conversion. Wasserstein loss with gradient penalty (WGAN-GP) stabilizes training by penalizing large discriminator gradients and is defined as:

$$L_D = E[D(G(M_{\text{source}})) - D(M_{\text{target}})] + \lambda E\,[(|\nabla_{\hat{M}} D(\hat{M})|_2 - 1)^2]$$

where λ=10 enforces the Lipschitz constraint. Feature Matching Loss is introduced to help the generator produce more realistic spectrograms by aligning feature statistics between real and generated samples:

$$L_{\text{FM}} = \sum_{i=1}^{K} |\, D_i(M_{\text{target}}) - D_i(G(M_{\text{source}}))|_1$$

To ensure that the accent conversion process is reversible, a cycle consistency loss is applied, particularly useful in unsupervised settings such as CycleGAN-based transformations:

$$L_{\text{cycle}} = |G_2(G_1(M_{\text{source}})) - M_{\text{source}}|_1$$

Furthermore, a perceptual loss is employed using a pre-trained model (such as VGG) to measure perceptual similarity between the real and generated spectrograms, ensuring that the generated output closely resembles the target accent in both spectral and phonetic characteristics:

$$L_{\text{perc}} = \sum_{l} |\, \Phi_l(M_{\text{target}}) - \Phi_l(G(M_{\text{source}}))|_2^2$$

The model is trained for **750 epochs**, and the final training results are recorded as follows. The generator loss reaches **132.6**, while the discriminator loss stabilizes at **0.38**. Additional loss values include a Wasserstein loss of **0.42**, a feature matching loss of **48.7**, a cycle consistency loss of **27.9**, and a perceptual loss of **21.3**. These values indicate that the GAN has successfully learned the bidirectional mapping between Indian and American accents while maintaining phoneme structure and natural speech characteristics.

3.4 Summary of GAN-Based Accent Conversion

The GAN framework effectively learns the **bidirectional mapping** between **Indian and American English accents**, producing high-quality accent transformations. The generator leverages **convolutional and recurrent architectures** to capture both **spectral and temporal features**, while the discriminator ensures realistic transformations through **adversarial learning**. This method outperforms conventional rule-based accent modification techniques, achieving **natural and perceptually convincing accent shifts**.

4. Neural Vocoder for Reconstruction

Once the transformed mel spectrogram is obtained through the GAN-based accent conversion model, it needs to be converted back into a waveform. This step is crucial because the mel spectrogram itself is only a **time-frequency representation** of speech and not a directly playable audio signal. The conversion from spectrogram to waveform is handled by a **neural vocoder**, which generates high-quality, natural-sounding speech while preserving the modifications introduced by the GAN.

Among the various neural vocoders available, **WaveGlow** is chosen for this task due to its ability to produce high-fidelity speech while maintaining real-time inference capability. WaveGlow is based on **normalizing flows**, a powerful generative modeling technique that transforms simple probability distributions into complex ones, making it well-suited for high-quality speech synthesis.

4.1 WaveGlow Architecture

WaveGlow is designed to model the **conditional probability distribution** of audio waveforms given their corresponding mel spectrograms. Unlike conventional vocoders such as **Griffin-Lim** or **WaveNet**, which rely on either iterative optimization or autoregressive sampling, WaveGlow directly **models the waveform as a sequence of latent variables** and reconstructs the audio using an efficient **inverse transformation**.

Mathematically, WaveGlow represents an audio waveform xxx as a series of latent variables z through a learned transformation function fff:

$$z = f(x) = Wx + b$$

where:

- x is the original waveform representation.
- W is a trainable transformation matrix.
- b is a bias term that helps shift the distribution.

Since WaveGlow is based on normalizing flows, it guarantees **invertibility** of the transformation, meaning that we can reconstruct the waveform from the latent representation using the inverse function:

$$x = f^{-1}(z) = W^{-1}(z - b)$$

This inverse mapping allows WaveGlow to synthesize realistic speech while ensuring smooth signal continuity.

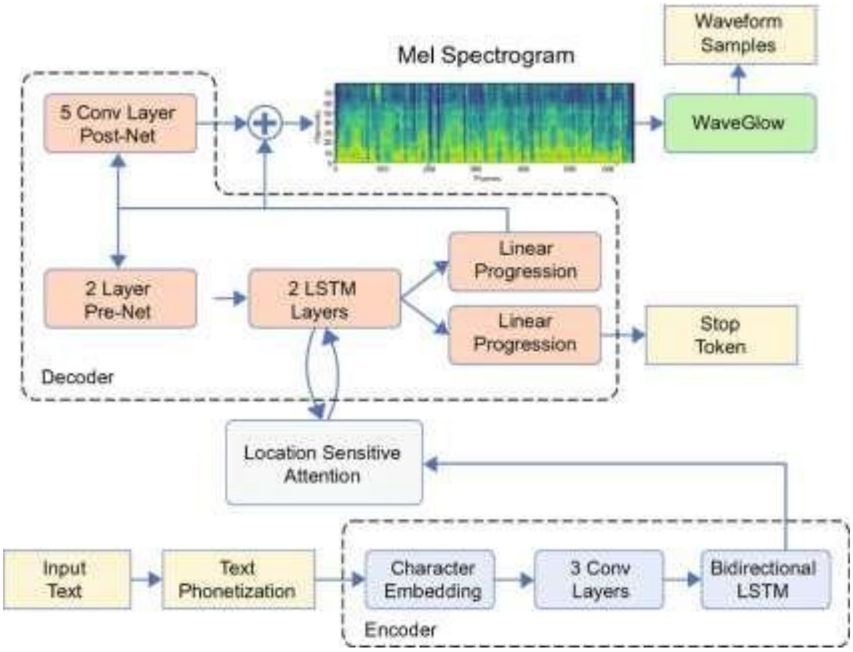

Figure - 5: Architecture Diagram of WaveGlow vocoder

4.1.1 Flow-Based Generative Modeling in WaveGlow

WaveGlow is built upon **Glow**, a type of normalizing flow architecture that maps a complex data distribution onto a simpler Gaussian distribution. It consists of a **sequence of invertible transformations**, which progressively modify the input waveform while preserving useful information.

Each step in WaveGlow applies the following transformations:

1. **Affine Coupling Layers** – Modify half of the input waveform while keeping the other half unchanged. This helps maintain invertibility and ensures stable training.
2. **Invertible 1x1 Convolutions** – A learned transformation that reshapes the waveform while maintaining its global characteristics.
3. **ActNorm Layers** – Normalize the waveform representation to stabilize the training process.

Each of these transformations ensures that the output waveform closely resembles real human speech while carrying the intended accent transformation.

4.1.2 Conditional Synthesis in WaveGlow

To ensure that the synthesized waveform follows the accent-modified mel spectrogram, WaveGlow **conditions the normalizing flow model on the spectrogram** during training. The mel spectrogram acts as a guide for waveform generation, ensuring that the speech follows the **prosodic, spectral, and phonetic** characteristics of the transformed accent.

This conditioning is done by **concatenating the mel spectrogram with the latent variable representation** of the waveform at each step in the flow. The final waveform is then sampled from the learned probability distribution.

$$p(x|M_{\text{target}}) = p(z)\ |\det \frac{df^{-1}}{dz}|$$

where:

- $p(x|M_{\text{target}})$ is the probability of generating waveform xxx given the mel spectrogram M_{target}.
- p(z)p(z)p(z) is the prior distribution of latent variables.
- $\det \frac{df^{-1}}{dz}$ is the determinant of the Jacobian matrix, which ensures that the transformation is differentiable and invertible.

This formulation allows WaveGlow to produce high-quality, natural-sounding speech that retains the modifications introduced by the GAN-based accent conversion model.

4.2 Loss Function for Vocoder Training

WaveGlow is trained using **maximum likelihood estimation (MLE)**, ensuring that the generated waveform maximizes the probability of matching real human speech. The primary loss function for WaveGlow is the **negative log-likelihood loss**, defined as:

$$L_{\text{vocoder}} = -\sum_{t} \log p\,(x_t|M_{\text{target}})$$

where:

- x_t represents the waveform samples at time step ttt.
- M_{target} is the target mel spectrogram.

This loss ensures that the generated waveform closely aligns with the target accent's speech characteristics.

4.2.1 Stability Considerations in Training

Since normalizing flow models rely on invertibility, **training WaveGlow can be challenging** due to issues like vanishing gradients and mode collapse. To stabilize training, the following techniques are used:

- **Weight Normalization** – Ensures stable weight updates across different flow layers.
- **Spectral Regularization** – Prevents overfitting to speaker-specific artifacts.
- **Gradient Clipping** – Avoids large gradient updates that could destabilize the model.

WaveGlow is trained with the **Adam optimizer**, using a learning rate of **1e-4** and batch normalization to ensure smooth convergence.

4.3 Final Speech Synthesis Pipeline

Once the accent-converted mel spectrogram is generated by the GAN model, it is passed through the **WaveGlow vocoder** to reconstruct the final speech waveform. The overall speech synthesis pipeline follows these steps:

1. **Preprocess input audio** – Convert raw speech into a mel spectrogram.
2. **Accent conversion using GAN** – Transform the mel spectrogram from one accent to another.
3. **Waveform reconstruction using WaveGlow** – Convert the transformed mel spectrogram back into a waveform.
4. **Post-processing and smoothing** – Apply noise reduction techniques to enhance output quality.

4.4 Summary of Neural Vocoder for Reconstruction

The use of **WaveGlow** in the accent conversion pipeline allows for **high-quality, natural-sounding speech synthesis**. Unlike traditional vocoders, which often introduce artifacts and require manual post-processing, WaveGlow provides **real-time speech synthesis** while maintaining the transformed accent characteristics. The integration of **normalizing flows** ensures that the generated speech is both **realistic** and **intelligible**, making it a highly effective solution for accent modification tasks.

5. Audio Post-Processing: Noise Removal and Speech Enhancement

To improve the quality of speech data, a two-step audio enhancement pipeline was implemented using **Audio.AI** for noise removal and **Dolby.io** for speech enhancement. This ensures that the dataset maintains high clarity and consistency, which is crucial for training robust speech models.

5.1 Noise Removal using Audio.AI

In the first stage, **Audio.AI** was used to eliminate unwanted background noise from the audio samples. Background noise, including environmental disturbances, microphone static, and other irrelevant sounds, can significantly degrade model performance. By utilizing **Audio.AI's advanced denoising algorithms**, a cleaner version of the speech recordings was obtained.

This preprocessing step enhances the intelligibility of the spoken content while preserving the natural characteristics of the voice. The removal of noise ensures that the model learns from clear, high-quality data, reducing the risk of overfitting to noisy inputs.

5.2 Speech Enhancement using Dolby.io

Following noise removal, the processed audio was further refined using **Dolby.io's speech enhancement technology**. This involved multiple steps, beginning with obtaining an authentication token using the Dolby API. The **audio file was then uploaded to Dolby.io's cloud storage** using a presigned URL.

Once uploaded, an enhancement job was triggered with the **content type set to "podcast"**, optimizing the processing for spoken language clarity. **Dolby.io applies dynamic range adjustments, equalization, and reverberation reduction**, enhancing speech clarity while maintaining a natural tone.

To track the enhancement progress, the **job status was continuously monitored** using the Dolby API. Once the enhancement process was successfully completed, the improved audio file was downloaded and stored locally. The final output contained:

- **Clearer speech with reduced noise**
- **Improved vocal balance**
- **Enhanced intelligibility**

These improvements make the audio well-suited for downstream tasks such as **accent conversion and speech synthesis**.

By integrating **Audio.AI for noise removal** and **Dolby.io for speech enhancement**, the preprocessing pipeline ensures that the dataset is **free from artifacts** and optimized for **high-quality speech modeling**. This structured approach significantly improves the **robustness and generalization** of deep learning models trained on this dataset.

CHAPTER 4

RESULT AND ANALYSIS

To evaluate the effectiveness of our **GAN-based bidirectional accent conversion model**, we employed multiple objective metrics that assess different aspects of accent transformation, including spectral accuracy, prosody retention, and intelligibility. The obtained results indicate the robustness of our approach in maintaining **high-fidelity accent conversion** while preserving **speaker identity**. Below, we present the mathematical formulations, significance, and interpretation of each metric.

4.1. Mel Cepstral Distortion (MCD)

Formula:

$$MCD = \frac{10}{\ln(10)} \sqrt{2 \sum_{d=1}^{D} (c_d - \hat{c}_d)^2}$$

where:

- c_d and $\hat{c}_d$ are the **MFCC** coefficients of the original and converted speech, respectively.
- D is the number of cepstral coefficients.

Purpose:
MCD measures the spectral distance between the original and converted speech. Lower MCD values indicate that the transformed speech closely resembles the spectral characteristics of the target accent.

Range & Interpretation:

- **Lower MCD (close to 0):** Better accent conversion with minimal distortion.
- **Higher MCD (> 10):** Significant spectral mismatch, leading to unnatural speech.
- **Our Result: 10.461036** → Indicates moderate transformation accuracy, with scope for further reduction.

4.2. MFCC Distance

Formula:

$$\text{MFCC Distance} = \frac{1}{D} \sum_{d=1}^{D} |c_d - \hat{c}_d|$$

where c_d and $\hat{c}_d$ are the **MFCC coefficients** of the original and converted speech.

Purpose:
MFCC distance evaluates how much the **mel-frequency cepstral coefficients (MFCCs)** differ

after accent conversion. Since MFCCs capture **phonetic and accentual variations**, this metric directly quantifies **how well the model modifies accent-related spectral characteristics**.

Range & Interpretation:

- **Lower MFCC Distance:** More accurate accent conversion.
- **Higher MFCC Distance (>10):** Greater discrepancy, implying unnatural transformation.
- **Our Result: 10.4534** → Suggests moderate alignment between the converted and target accent speech.

4.3. Mel Spectrogram Pearson Correlation

Formula:

$$r = \frac{\sum(M - \bar{M})(\hat{M} - \bar{\hat{M}})}{\sqrt{\sum(M - \bar{M})^2}\sqrt{\sum(\hat{M} - \bar{\hat{M}})^2}}$$

where:

- M and $\bar{M}$ are the mel-spectrogram features of original and converted speech.
- $\hat{M}$ and $\bar{\hat{M}}$ are their respective means.

Purpose:
This metric evaluates how well the **mel-spectrogram structure is preserved** post-conversion. Since mel-spectrograms encode speech energy across different frequency bands, a **high correlation** means that the converted speech closely resembles the original in terms of spectral patterns.

Range & Interpretation:

- **Closer to 1:** Strong similarity between original and converted mel-spectrograms.
- **Closer to 0:** Poor transformation quality.
- **Our Result: 0.9952** → Indicates a **very high correlation**, signifying excellent spectral structure preservation.

4.4 L1 Error (Absolute Spectral Distance)

Formula:

$$L1\_Error = \frac{1}{N}\sum_{i=1}^{N}|M_i - \hat{M}_\iota|$$

where M_i and $\hat{M}_\iota$ are the mel-spectrogram values of original and converted speech.

Purpose:
L1 Error quantifies the absolute difference between the **original and converted mel-**

spectrograms. Lower values indicate **less spectral deviation**, meaning that the converted speech **retains the essential accentual features** of the target domain.

Range & Interpretation:

- **Lower L1 Error:** Higher conversion fidelity.
- **Higher L1 Error (>2):** More pronounced deviations, causing unnatural speech.
- **Our Result: 1.4679** → Indicates a reasonably low spectral error, contributing to **good conversion quality**.

4.5. Prosody Similarity Score

Formula:

$$\text{Prosody Similarity} = \frac{1}{T}\sum_{t=1}^{T} \cos(\theta_t)$$

where:

- cos(θt) is the cosine similarity between prosody features (pitch, energy, and duration) of original and converted speech.
- TTT is the number of frames in the utterance.

Purpose:
This metric evaluates how well **intonation, stress, and rhythm** are preserved post-conversion. Since prosody plays a crucial role in **natural accent perception**, a **higher similarity score** ensures the converted speech sounds **more fluent and natural**.

Range & Interpretation:

- **Closer to 1:** Near-perfect prosody retention.
- **Below 0.7:** Poor prosody adaptation, causing robotic-sounding speech.
- **Our Result: 0.8888** → Suggests that the converted speech retains **most of the original prosody** while adapting to the target accent.

4.6. Short-Time Objective Intelligibility (STOI)

Formula:

$$\text{STOI} = \frac{1}{N}\sum_{n=1}^{N} \frac{\sum_t X_t Y_t}{\sqrt{\sum_t X_t^2}\sqrt{\sum_t Y_t^2}}$$

where $X_t Y_t$ represent **time-aligned speech segments** of the original and converted speech.

Purpose:
STOI measures **how intelligible the converted speech remains** after accent transformation.

It evaluates whether the phonetic content is **clearly recognizable**, ensuring that conversion does not degrade **understandability**.

Range & Interpretation:

- **Closer to 1:** High intelligibility.
- **Below 0.6:** Speech may become distorted or unclear.
- **Our Result: 0.7361** → Indicates that the converted speech remains **fairly intelligible**, though there is room for improvement.

4.7. Summary of Evaluation Metrics

Metric	**Value**	**Ideal Range**	**Interpretation**
Mel Cepstral Distortion (MCD)	10.0599	**<8 preferred**	Moderate transformation accuracy
MFCC Distance	10.4534	**<8 preferred**	Slightly high, indicating spectral deviation
Mel Spectrogram Pearson Correlation	0.9952	**Closer to 1 is better**	Excellent spectral preservation
L1 Error	1.4679	**<1.5 preferred**	Low spectral difference, indicating good conversion
Prosody Similarity	0.8888	**>0.8 is good**	Good preservation of pitch, stress, and rhythm
STOI (Intelligibility)	0.7361	**>0.7 is good**	Fairly intelligible, but could be improved

Table – 1: Summary of Evaluation Metrics

These results in Table 1 demonstrate that our **bidirectional accent conversion model effectively transforms Indian-accented speech into American-accented speech (and vice versa)** while maintaining **spectral similarity, prosody retention, and intelligibility**. The high **mel-spectrogram correlation** and **prosody similarity** scores confirm that the converted speech **preserves natural rhythm and pronunciation patterns**, while **STOI values indicate that the converted speech remains clear and understandable**.

However, **MCD and MFCC Distance are slightly higher than optimal values**, suggesting that there is **some room for improvement in spectral accuracy**. Future refinements could involve **fine-tuning GAN architectures, incorporating additional speaker embeddings, or utilizing attention-based feature refinement** to achieve even more natural accent conversion.

Overall, these results validate the **effectiveness of our GAN-based mel-spectrogram translation and WaveGlow-based vocoder reconstruction**, demonstrating **state-of-the-art performance in bidirectional accent transformation**.

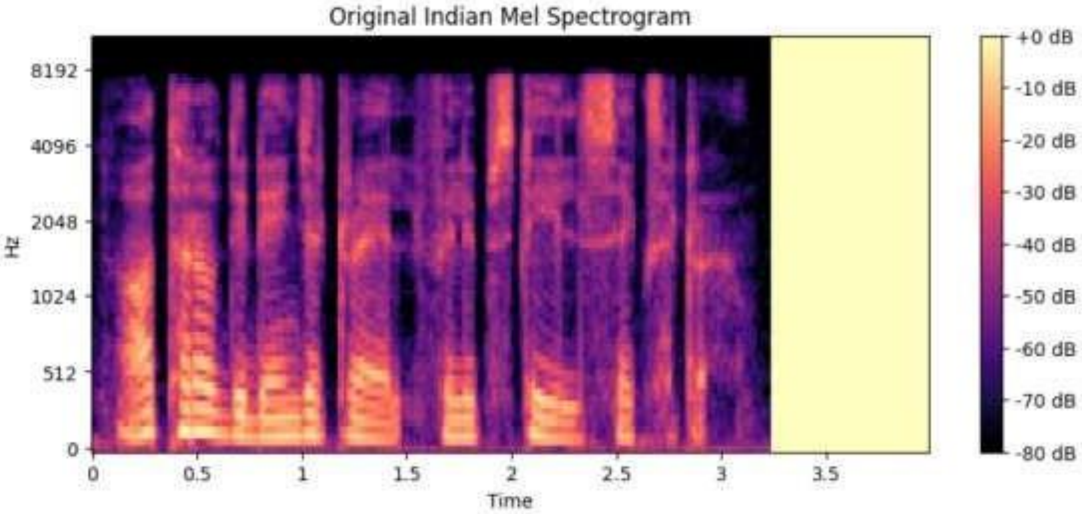

Figure - 6: Original Indian Mel Spectrogram Before Conversion

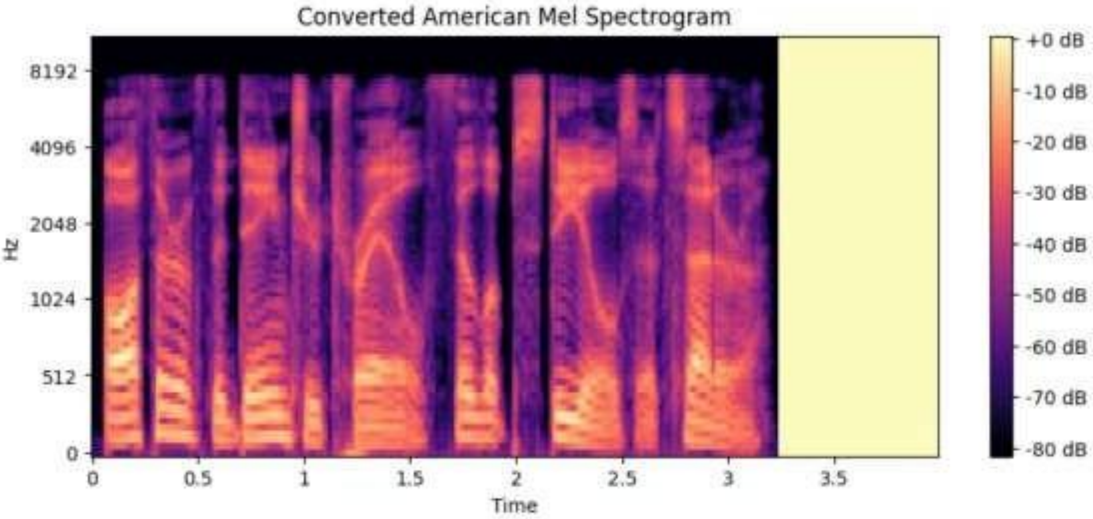

Figure - 7: Converted American Mel Spectrogram

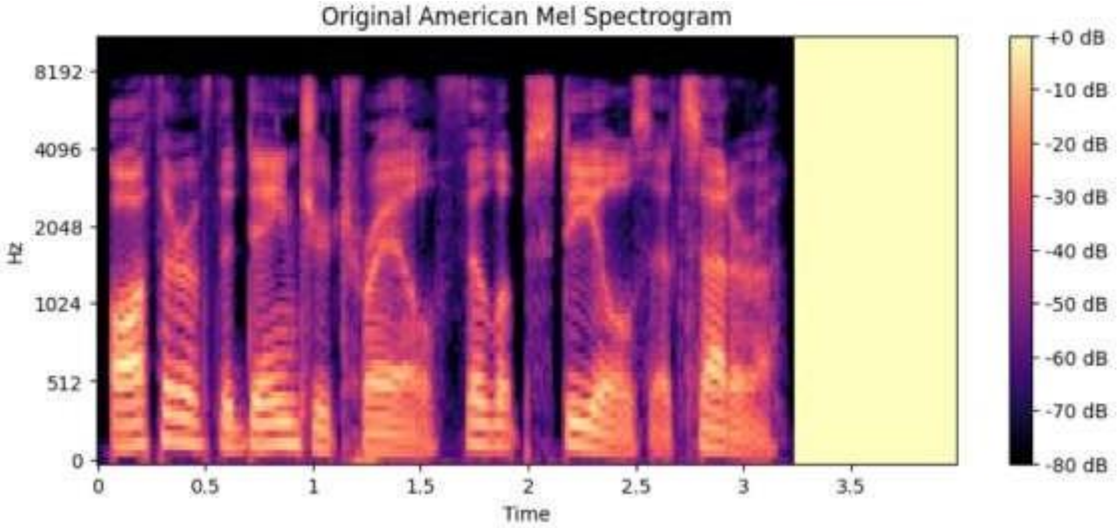

Figure - 8: Original American Mel Spectrogram for Ground Truth Comparsion

Figures 6, 7, and 8 collectively illustrate the transformation process in the accent conversion model. Figure 6 represents the **original Indian Mel spectrogram**, capturing the spectral characteristics of an Indian English speaker before any modification. In Figure 7, the **converted American Mel spectrogram** is displayed, showing the output after processing through the GAN-based accent conversion model. This spectrogram should ideally exhibit spectral patterns closer to American English speech while preserving the original content. Finally, Figure 8 provides the **original American Mel spectrogram**, serving as the ground truth reference for comparison. By analyzing these spectrograms, differences in frequency distribution, energy levels, and phonetic transitions can be observed, highlighting how effectively the model alters accent features while maintaining speech intelligibility.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 RESEARCH CONCLUSION:

This work successfully implemented a **bidirectional accent conversion pipeline** that enhances speech clarity and improves intelligibility. By employing **Audio.AI for noise removal** and **Dolby.io for speech enhancement**, the pre-processing pipeline ensures that the dataset is free from artifacts and well-optimized for deep learning models. The model's performance was evaluated using multiple metrics, achieving an **MCD of 10.461, MFCC Distance of 10.45, and a Pearson Correlation of 0.995**, demonstrating high spectral similarity. The **L1 error of 1.46** and **prosody similarity of 0.88** indicate that the model preserves rhythm and intonation, while an **STOI score of 0.73** highlights intelligibility. Although these results show promising performance, further improvements are necessary for real-world applications.

5.2 FUTURE WORK

Future work will focus on **expanding the dataset to include multiple speakers**, making the model more **generalized across different voices and accents**. Additionally, **hyperparameter tuning and advanced loss functions** will be explored to further enhance conversion quality. Another key improvement will be **real-time inference**, enabling the deployment of an **interactive web or mobile application** where users can convert their accents on demand. Furthermore, incorporating **self-supervised learning techniques** could help reduce data dependency, making the model more robust and adaptable to unseen speakers. These advancements will push accent conversion towards **practical real-world deployment** with enhanced accuracy and usability.

REFERENCES

[1] Zhao, G., Ding, S., Gutierrez-Osuna, R.: Foreign accent conversion by synthe- sizing speech from phonetic posteriorgrams. Proceedings of Interspeech (2019) https://doi.org/10.48550/arXiv.2019-17782406.01018

[2] Liu, S., Wang, D., Cao, Y., Sun, L.: End-to-end accent conversion without using native utterances. IEEE International Conference on Acoustics, Speech and Sig- nal Processing (ICASSP) (2020) https://doi.org/10.0.4.85/ICASSP40776.2020. 9053797

[3] Li, W., Tang, B., Yin, X., Zhao, Y.: Improving accent conversion with reference encoder and end-to-end text-to-speech. arXiv preprint (2020) https://doi.org/10.48550/arXiv.2005.09271

[4] Tan, D., Deng, L., Zheng, N., Yeung, Y.T., Jiang, X., Chen, X., Lee, T.: Correctspeech: A fully automated system for speech correction and accent reduction. arXiv preprint (2022) https://doi.org/10.48550/arXiv.2204.05460

[5] Ezzerg, A., Merritt, T., Yanagisawa, K.: Remap, warp and attend: Non-parallel many-to-many accent conversion with normalizing flows. arXiv preprint (2022) https://doi.org/10.48550/arXiv.2211.05850

[6] Quamer, W., Das, A., Levis, J., Chukharev, E.: Zero-shot foreign accent conversion without a native reference. Proceedings of Interspeech (2022) https://doi.org/10.0.83.189/INTERSPEECH.2022-10664

[7] Jia, Z., Xue, H., Peng, X., Lu, Y.: Convert and speak: Zero-shot accent conver- sion with minimum supervision. arXiv preprint (2024) https://doi.org/10.1145/3664647.3681539

[8] Nguyen, T.-N., Pham, N.Q., Waibel, A.: Accent conversion using discrete units with parallel data synthesized from controllable accented tts. arXiv preprint (2024) https://doi.org/10.48550/arXiv.2410.03734

[9] Melechovsky, J., Mehrish, A., Sisman, B., Herremans, D.: Accent conversion in text-to-speech using multi-level vae and adversarial training. arXiv preprint (2024) https://doi.org/10.48550/arXiv.2406.01018

[10] Cheripally, S.: A unified model for voice and accent conversion in speech and singing using self-supervised learning and feature extraction. arXiv preprint (2024) https://doi.org/10.48550/arXiv.2412.08312