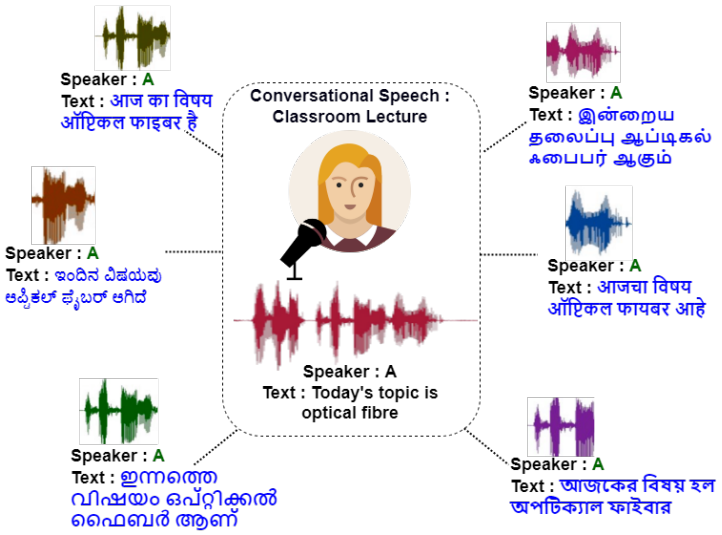
DEPARTMENT OF COMPUTER
SCIENCE & ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY
MADRAS
CHENNAI - 600036

Towards Cross-lingual Voice Adaptation for Conversational Speech



A Thesis

Submitted by

BHAGYASHREE MUKHERJEE

For the award of the degree

Of

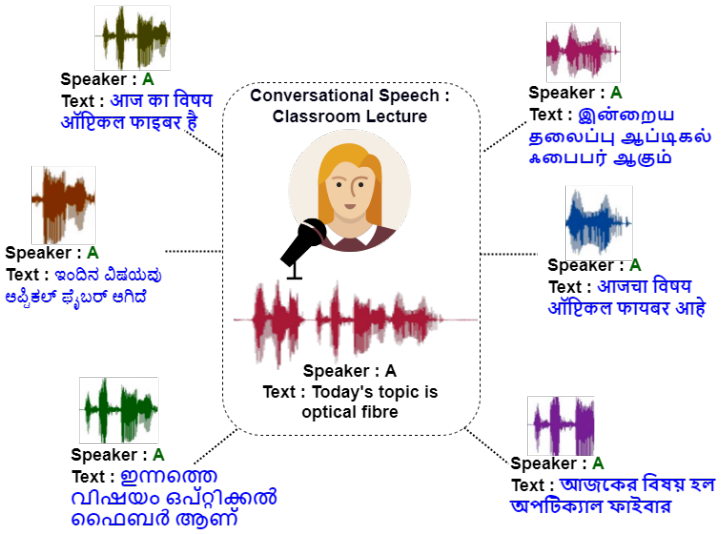
MASTER OF SCIENCE

by Research

January 2023

DEPARTMENT OF COMPUTER
SCIENCE & ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY
MADRAS
CHENNAI - 600036

Towards Cross-lingual Voice Adaptation for Conversational Speech



A Thesis

Submitted by

BHAGYASHREE MUKHERJEE

For the award of the degree

Of

MASTER OF SCIENCE

by Research

January 2023

THESIS CERTIFICATE

This is to undertake that the Thesis titled **Towards Cross-lingual Voice Adaptation for Conversational Speech**, submitted by **Bhagyashree Mukherjee**, to the Indian Institute of Technology Madras, for the award of M.S., is a bonafide record of the research work done by me under the supervision of Prof. Hema A Murthy. The contents of this Thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Place: Chennai 600036

Date: January 2023

Bhagyashree Mukherjee

Research Scholar

Guide

Prof. Hema A Murthy

LIST OF PUBLICATIONS

PUBLICATIONS IN CONFERENCE PROCEEDINGS

1. Bhagyashree Mukherjeee, Anusha Prakash, and Hema A. Murthy. "Analysis of Conversational Speech with Application to Voice Adaptation." In 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2021, pp. 765-772, doi:10.1109/ASRU51503.2021.9688146.

ACKNOWLEDGEMENTS

As I come towards the end of my Master's life at IIT Madras, I am filled with gratitude to everyone who supported me in my journey in a big or a small way. First of all, I would like to thank my guide, Prof. Hema A Murthy, for being one of the most supportive person in my IIT life. I feel incredibly privileged and honored to work under her guidance. Her experience and expertise in research have always nurtured my interest in topics in which I was a novice. Her timely advice and support have helped me to overcome minor setbacks and be confident even at difficult times. Her dedication to her work, intertwined with her incredible ideas to use research for the betterment of society, will be an inspiration throughout my life. I am very grateful to my Graduate Test Committee members - Prof. Krishna M. Sivalingam, Prof. Umesh S., and Prof. Arun Rajkumar for monitoring my progress in research.

I express my sincere gratitude to Karthik, Navina, Arun Baby, Anusha, Jom, Anand, and Gowriprasad for supporting me personally and professionally throughout my research. I thank my friends Amit, Ishika, Mano, Saish, and Sudhanshu for making these three years memorable and cheerful. I want to thank other team members - Mohana ma'am, Ashish, Vasista, Arun, Nithya, Vrunda, Prabhakaran, and Metilda for their help and support. I would also like to thank Jeyanti ma'am for taking care of the administrative procedures of my projects.

My dream of pursuing Masters from IIT Madras would have been incomplete without the support of my family. I am very grateful to my parents, Mr. Harendra Nath Mukherjee and Mrs. Jaba Mukherjee, who have been the pillars of my strength in every phase of my life. Special thanks to my sister Piyali, who has always been there for me in good and bad times.

Bhagyashree Mukherjee

ABSTRACT

KEYWORDS: Conversational speech, voice adaptation, analysis, speech synthesis, classroom lectures

Voice adaptation and conversion have especially gained impetus in the personification of speech-enabled systems, movie dubbing, lecture dubbing, singing voice transformation, and voice adaptation for speech disordered patients. The accessibility of mobile phones in interior regions in India and numerous online educational content have led to the demand for lectures being available in regional languages. While transcreation of lecture videos in a number of different languages is a tall order, an attendant problem is the transcreation of the video in the original speaker's voice. The voice synthesized in the target language needs to be matched to that of the source voice. This is difficult even for read speech but becomes even more complex for conversational speech. We examine classroom lectures with the objective of dubbing lectures from English to various Indian languages. The task is challenging as there are many problems associated with it. Firstly, classroom lectures are essentially conversational with fluctuations in speaking rate and contain disfluencies due to typical speaker mannerisms. Secondly, we attempt to perform cross-lingual voice transformation from English to Indian languages (e.g., Hindi, Kannada), which are phonotactically very different.

Most speech synthesis and voice conversion systems are trained on "read-speech," where the speech is rehearsed, unlike conversational speech, which is spontaneous. We analyze why Text-to-Speech (TTS) synthesis systems, which produce highly intelligible and robust audios for read speech, fail to model conversational speech. We compare read speech and conversational speech with respect to pitch variation, syllable rate variation, and signal-to-noise ratio (SNR) and identify the differences. Due to the lack of a conversational multispeaker dataset, we create our own dataset for the analysis task. Since the lecture transcriptions are generated by an Automatic Speech Recognition (ASR) model and manual curation is cumbersome, we devise data pruning techniques to curate the data and use this data to train a TTS model.

Further, to achieve the objective of dubbing lectures from English to Indian languages, a bilingual (Indian language + English) text-to-speech model trained on read speech is adapted to the required speaker's voice using a minimally transcribed lecture recording. The novelty of this work lies in adapting read speech models using conversational speech data to generate the target speaker's voice. The ASR-generated transcriptions are manually curated to maintain accurate text-audio correspondence. Two different frameworks have been used for adaptation – HTS (HMM-based speech synthesis system), a statistical parametric model, and E2E (End-to-End), a neural network-based model. X-vectors are used as speaker embeddings in the E2E framework to enhance speaker characteristics. The analysis and findings pave the way for further exploration of conversational TTS, cross-lingual voice adaptation, and voice conversion in a low-resource scenario.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ASR	Automatic Speech Recognition
CLS	Common Label Set
DMOS	Degraded Mean Opinion Score
DNN	Deep Neural Network
DTW	Dynamic Time Warping
E2E	End-to-End
GAN	Generative Adversarial Network
GD	Group Delay
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
HTS	Hidden Markov Model Speech Synthesis System
LMR	Linear Multivariate Regression
LPC	Linear Predictive Coefficient
MAP	Maximum a posteriori
MCD	Mel-Cepstral Distortion
MFCC	Mel-Frequency Cepstral Coefficient
MLPG	Maximum Likelihood Parameter Generation
MT	Machine Translation
S2S	Speech-to-Speech
SNR	Signal-to-Noise Ratio
SRT	SubRip Text
STE	Short Term Energy
TTS	Text-to-Speech
UBM	Universal Background Model
USS	Unit Selection Synthesis
VAD	Voice Activity Detection
VC	Voice Conversion
WER	Word Error Rate

CHAPTER 1
INTRODUCTION

Speech synthesis research has made remarkable advancements in recent years. Statistical models such as Hidden Markov Model (HMM) based speech synthesis using STRAIGHT (Kawahara, 2006) vocoder are now capable of synthesizing audio equivalent to that of human-produced speech. End-to-End (E2E) systems, the current state-of-the-art in speech synthesis, are at par or even better than statistical models. E2E systems take into account the entire sentence while synthesizing speech. This leads to good prosody across the whole sentence. However, the performance of these speech synthesis systems is limited to the reproduction of read speech. Training such systems with conversational speech leads to poor synthesis owing to the disfluencies in conversations and unstructured sentences. The fluctuations in speaking rate and pitch, abrupt pauses, and incomplete sentence endings pose difficulty in training a conversational speech synthesis model. There are also no standard conversational speech datasets. Neural networks require a huge amount of clean data for training which adds to the challenge. Although attempts have been made to use conversational speech data in automatic speech recognition and machine translation tasks, very few attempts have been made to build text-to-speech models from scratch using conversational speech.

Most real-world speech applications like chatbots and voice assistants use conversational speech to facilitate better user-computer interactions. Hence, it is essential to understand the fundamental differences between read speech and conversational speech and deploy technologies to develop robust conversational TTS systems. Taking a further step towards conversational speech research, we explore conversational speech cross-lingual voice adaptation. This is furthermore challenging due to the vast variations between the speaking styles of different speakers. Imbibing a person's voice in a different language poses severe difficulty due to the variations in the phonotactics of the two languages. In this thesis, we use open-source online educational lectures as conversational speech datasets and attempt to generate the lecturer's voice in a target Indian language.

This chapter is organized as follows: Section 1.1 discusses the motivation for pursuing cross-lingual voice adaptation for conversational speech. Section 1.2 digresses into the different applications of conversational speech voice adaptation. The overview of the thesis is discussed in Section 1.3. Section 1.4 and Section 1.5 encompasses the major contributions and the organization of the thesis respectively.

1.1 MOTIVATION

The rapid technological drift towards online educational platforms has led to the availability of numerous educational videos online. These course lecture recordings are open-source and easily accessible forms of conversational speech data. Most of these lectures are available in English. The transcreation of these videos in various Indian languages can help reach many target audiences, even in the interior regions of our country. According to a survey conducted in India, about 75% of students receive their primary and secondary education in the vernacular language (AiutoConsulting, 2020). At the same time, most of the undergraduate courses are taught in English. Undergraduate lecture availability in various Indian languages can be an initiative to alleviate the language barrier between English and Indian languages. To present a real-time scenario and enhance the user experience, the original speaker's voice in the dubbed videos will sound more natural. With this objective of dubbing technical lectures from English to Indian languages, we explore classroom lectures which are conversational and spontaneous. Two motives support the objective - enhancing naturalness in the synthesized audios and imparting the speaker's voice characteristics using a small amount of data.

A typical dubbing system involves recognizing the original speech using Automatic Speech Recognition (ASR), translating the recognized text into the required native language using Machine Translation (MT), synthesizing the translated text, and voice adaptation to generate the synthesized sentences in the speaker's voice and to sync the synthesized audio with the original video (Figure 1.1). The task is difficult as class lectures are extempore, with prosodic variations and mannerisms inherent to a particular speaker that lead to disfluencies. Hence, preserving the speaker's voice in a cross-lingual setting (English to Indian languages) becomes more challenging.

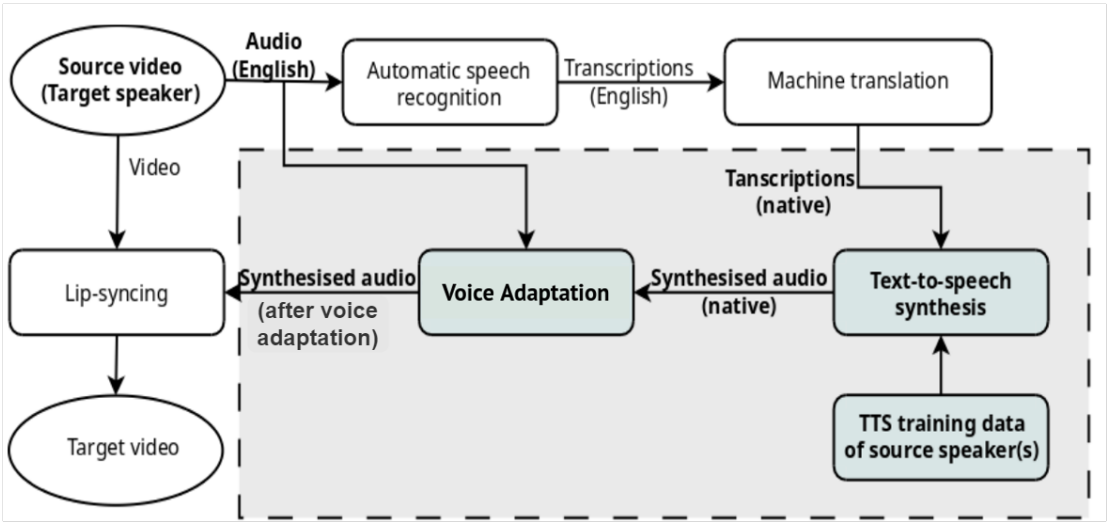


Fig. 1.1: Flowchart of a typical dubbing system. The highlighted block is the voice adaptation module.

1.2 APPLICATIONS

Apart from lecture dubbing, there are several other applications of conversational speech voice conversion and adaptation.

Real-time dialogue conversion

As shown in Figure 1.2, we can convert the content spoken by the general public to sound like Amitabh Bachchan. Dialogue-conversion systems are widely used in movie dubbing by preserving the voice of the original actor to present a better audience experience. It is also used for creating a voice-over for characters such as Doraemon and Donald Duck to sound like cartoon voices.

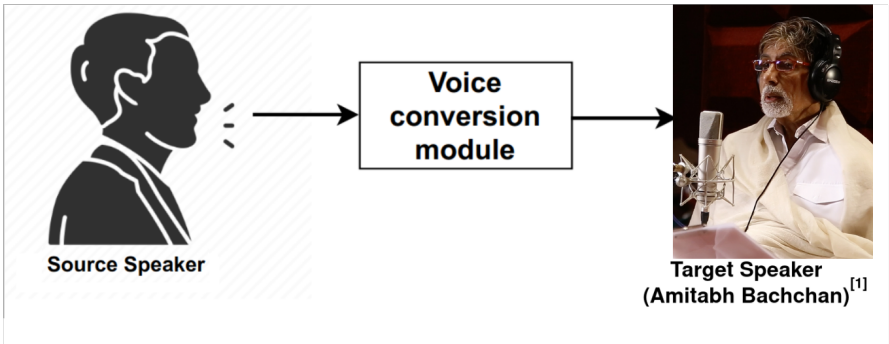


Fig. 1.2: Real-time dialog conversion

[1] Image Courtesy: Wikimedia Commons, CC-BY-SA-2.0

Voice adaptation for speech disorder patients

Many people suffer from speech disorders when the larynx (voice box) does not work due to laryngeal cancer, radiation exposure, or injury. Even in dysarthria, patients suffer from motor speech. In such cases, voice adaptation techniques can be applied to generate the speech in the original speaker's voice and help in better communication.

Singing voice transformation

The application of voice conversion can be extended to singing voice transformations. Songs sung by one artist can be converted to voices of different artists and in a different language using different voice adaptation techniques.

1.3 OVERVIEW OF THE THESIS

This thesis encompasses three aspects of speech technology - conversational speech, voice adaptation, and speech synthesis in Indian languages. First, it aims at understanding the fundamental difference between read-speech – which is rehearsed or scripted and conversational speech – which is spontaneous and unrehearsed. An extensive study is performed to demonstrate the variability in the speaking styles of different individuals while delivering a class lecture. An attempt is made to build a conversational text-to-speech synthesis model. Since classroom lectures are extempore, the task becomes more challenging due to the lack of manually curated transcriptions. The transcriptions are generated using automatic speech recognition models, which may be inaccurate. Other factors also influence training a TTS directly from lecture data; hence, pruning techniques are employed to curate the data to make it suitable for building robust conversational speech TTS systems.

The ultimate objective is to generate the target speaker's (lecturer) voice in any Indian language using minimally transcribed English data. Owing to the lack of Indian language data in the target speaker's voice, we propose to adapt a read-speech model using the conversational speech of the required speaker. The task is challenging because of the absence of manually annotated data for a particular speaker. The first part of the work discusses the issues in training a conversational text-to-speech synthesis system. The second part discusses two different speaker adaptation approaches using conversational speech data - one in the Hidden Markov Model-based speech synthesis

system (HTS) framework and the other in the End-to-End (E2E) framework using read-speech to train the initial model. Further, the adapted systems are compared, and the results are discussed.

1.4 CONTRIBUTION OF THE THESIS

The major contributions of this thesis are as follows :

- Analyzing the challenges involved in dealing with conversational speech when compared to read-speech
- Proposing pruning techniques to curate conversational speech data for training and adaptation
- Attempting cross-lingual voice adaptation to Indian languages using a small amount of speaker's data in a conversational scenario

1.5 ORGANIZATION OF THE THESIS

This thesis deals with –

- conversational speech
- voice-adaptation and voice-conversion
- speech synthesis in the context of Indian languages

Chapter 2 discusses the related literature in ASR and TTS using conversational speech. Further, an overview of different types of voice conversion and the state-of-the-art techniques are discussed. We further dive into the technologies used for speech synthesis and related work in the field of speech synthesis for Indian languages and discuss the paradigms used in the thesis.

In Chapter 3, an attempt is made to understand the differences between read speech and conversational speech. We discuss the issues regarding syllable rate, pitch, Signal-to-Noise Ratio (SNR), and disfluencies. We also indicate the possible errors in ASR transcripts. A comparative study of read speech and conversational speech from different speakers is attempted. Further, techniques for pruning the data are proposed to eliminate the manual effort and build intelligible and natural-sounding TTS. A conversational TTS system is trained using lecture data and compared with a TTS built using read speech data.

In Chapter 4, we discuss the phonotactic differences between English and Indian languages, which pose difficulty in cross-lingual voice adaptation. Handling of multilingual text and parsing techniques are discussed. Further, we propose adapting read-speech models using a small amount of lecture data to generate speaker characteristics. Two different speaker adaptation techniques are attempted - one in the Hidden Markov Model (HMM) speech synthesis system (HTS) and the other in an End-to-End (E2E) framework. The trained and adapted systems are compared, and the evaluation results are discussed. The extension of the techniques to multiple speakers and multiple languages is explored.

The summary of the thesis, scope for improvement, and future research directions are stated in Chapter 5.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 INTRODUCTION

Voice conversion has especially gained impetus in the context of online learning. With the outburst of the Covid-19 pandemic and numerous freely available online educational lectures, there has been an increased demand for technical lecture transcreation in Indian languages. We focus on the problem of imparting the target speaker's voice in an Indian language using classroom lectures that are in English. Classroom lectures are spontaneous and impromptu in contrast to read speech, which is scripted. We use open-source online technical lectures as a conversational speech dataset. The work deals with three aspects of speech technology — conversational speech in the form of classroom lectures, voice conversion and adaptation to the target speaker's voice, and speech synthesis for Indian languages to generate the target voice in a different language. Hence, the related work in all these three research directions is discussed.

In this chapter, Section 2.2 discusses different speech recognition and synthesis attempts using conversational speech. We review other techniques that have been applied in speech recognition to improve the error rate in the case of spontaneous speech recognition, handling disfluencies, and generating conversational-like voices. Conversational speech synthesis using manually curated data in a statistical and neural framework is explained. Section 2.3 elaborates different voice conversion and adaptation approaches. The incremental development in the domain of voice conversion, starting from spectral mapping (Desai *et al.*, 2010) to non-parallel and cross-lingual voice conversion, is discussed. Further, different techniques that are needed for speech synthesis in Indian languages and the approaches used in this thesis are stated in Section 2.4 and Section 2.5 respectively. The summary of the chapter is given in Section 2.6.

2.2 CONVERSATIONAL SPEECH : AN OVERVIEW

Conversational speech recognition and synthesis have always been a great challenge to the speech research community. The distinct speaker mannerisms, prosodic modulations, and syntactic conventions during conversations are very different from that of read speech which is just read aloud from a written transcription. Although conversational systems form the most important real-time application of speech, very few attempts have been made to build a text-to-speech (TTS) synthesizer using conversational data. This is because of the lack of hand-annotated conversational corpus, which is necessary for building robust models for recognition and synthesis tasks. Disfluencies, which are inherent to spontaneous speech, add further to the difficulty of the problem.

Several techniques have been proposed in the context of conversational speech recognition and segmentation. Meteer and Iyer (1996) use Switchboard Corpus with annotated disfluencies and filler words. They attempt to answer how conversational structure differs from the formal written structure and how these differences can be incorporated into the language model to improve speech recognition tasks. Dufour *et al.* (2009) present acoustic and linguistic cues for identifying spontaneous speech segments from an extensive audio database using the underlying characteristics of conversational speech. Rangarajan and Narayanan (2006) characterize the issue of repetitions in spontaneous speech and address the problem by building multiword level models for modeling the repetitions along with acoustic prosodic modeling. This helps in improving the ASR transcription error in the case of decoding conversational speech having multiple repetitions. Nanjo and Kawahara (2003) discuss unsupervised speaker-specific language model adaptation to handle the pronunciation variation in the case of spontaneous speech recognition. This is very vital to account for speaker variability to improve transcription accuracy.

The research in building text-to-speech synthesizers using conversational data is still at its inception because the filler pauses in conversations affect the model intelligibility. Still, attempts have been made to generate conversation-like voices as it is more expressive and enhances interactability. Sundaram and Narayanan (2003)

mention an empirical text processing method before speech synthesis using parts of speech (POS) taggers to generate spontaneous speech. However, this does not make use of conversational speech data. Andersson *et al.* (2012) present the idea of using conversational data for building an HMM-based speech synthesis model and attempts to incorporate the conversational phenomena in synthesized speech. Despite poor alignment at the phoneme level and variability in context, the synthesized audios sounded natural. A study by Székely *et al.* (2017) discusses building a deep neural network (DNN) speech synthesizer using conversational speech, which allows elongation of syllables, insertion of disfluencies, and voice modulation controls. The study indicates the effect of added disfluencies as perceived by a listener. However, both the works rely on a small amount of hand-annotated data where the filler pauses, disfluencies, and sentence-ending are marked carefully. Székely *et al.* (2019*b*) suggests that conversational speakers can be categorized into distinct breath groups. Breaths highly correlate to prosody and are agnostic to language or transcriptions. Breath duration can be exploited to segment audios and annotate a single speaker's conversational data, which can be further used for building robust TTS models. Székely *et al.* (2019*a*) uses podcast recordings, segments it using breath detectors, and transcribes automatically using Google Cloud API. As suggested by Baumann *et al.* (2017), since Google Cloud API omits filler pauses, disfluencies, and hesitations, IBM Watson Speech to Text API was used. Pronunciation and conversational characteristics were evaluated to show what is achievable in the field of conversational TTS. Baumann *et al.* (2017) describes the potential challenges in transcribing a dialogue system as the utterances may not be semantically correct.

2.3 VOICE CONVERSION AND ADAPTATION : AN OVERVIEW

Voice conversion (VC) is an important area of research in the speech domain. Given a source and a target speaker's speech signal, the task is to convert the audio signal from the source voice to the target by preserving the target's voice characteristics and the source's linguistic content. Speaker adaptation is a type of voice conversion wherein a pre-trained model is fine-tuned to generate the target speaker's voice. Based on the availability of the training data, voice conversion can be categorized into – parallel voice

conversion and non-parallel voice conversion.

In the case of parallel VC, the same utterances are present for the source and the target speakers. The availability of parallel data facilitates finding a mapping function between the source and the target acoustics. Several works have been attempted in the literature on parallel voice conversion. Parallel corpora VC techniques include vector-quantization (Abe *et al.*, 1988), spectral mapping (Desai *et al.*, 2010; Toda *et al.*, 2005), Gaussian Mixture Models (GMMs) followed by Dynamic Time Warping (DTW) (Toda *et al.*, 2001), Maximum Likelihood Parameter Generation (MLPG) (Toda *et al.*, 2005) and Linear Multivariate Regression (LMR) (Valbret *et al.*, 1992) to understand the dynamic trajectory. Abe *et al.* (1988) presents the idea of finding individual codebooks for source and target. Spectrum parameters, power values, and pitch frequencies are quantized separately. Further, DTW is performed for the parallel utterances, and histogram correspondence for codebook vectors of the target is computed as a linear combination of each codebook vector of the source. Stylianou *et al.* (1998) discuss finding a conversion function for the target using GMMs of the source speaker's spectral envelope. Mel-frequency Cepstral Coefficient (MFCC) features are used as they are decorrelated and hence do not degrade the performance even on using diagonal covariance matrices. Statistical parametric models like GMMs allow acoustic space modeling by pooling the source and target data into one codebook and finding a mapping between them. Spectral mapping between source and target speakers is performed using a Gaussian mixture model (GMM) of the joint probability distribution of the two speakers (Toda *et al.*, 2005). The technique is based on maximum likelihood estimation of spectral trajectory and outperforms frame-by-frame mapping techniques. Attempts have been made in the VC domain by reconstruction using framewise mapping of Linear Predictive Coefficient (LPC) features (Kain and Macon, 1998) and modifying pitch range to match the target. However, this was performed at the diphone level, not the utterance level.

The major challenge in VC is the availability of a limited amount of parallel corpora between the source and the target for learning the phonetic and prosodic mapping between the two speakers. Due to this limitation, exhaustive research has been carried out in non-parallel VC. Zhu and Yu (2012) proposes the idea of building

a Universal Background Model (UBM) by pooling all the speakers' data and then performing maximum a posteriori (MAP) adaptation to obtain the transformation function. Deep learning methods like speaker disentanglement (Chou *et al.*, 2018), one-shot voice conversion (Chou and Lee, 2019), and various other encoder-decoder architectures have been proposed in the literature to achieve natural-sounding speech. In speaker disentanglement, the linguistic and the acoustic features are disentangled. During synthesis, the target acoustic features and the source linguistic features are combined together to generate the speech in a target voice. Cross-lingual VC, i.e., changing the target speaker's language, adds to the difficulty of non-parallel VC since different languages have different phonetic representations. Cross-lingual VC approaches include bilingual phonetic posteriorgram and averaged models (Zhou *et al.*, 2019), Generative Adversarial Network (GAN) (Sisman *et al.*, 2019), frame alignment methods (Erro and Moreno, 2007) and variational autoencoders (Mohammadi and Kim, 2018). Bilingual and code-switched TTS is trained by Zhao *et al.* (2020) to perform cross-lingual VC using Mandarin and English data. The underlying approach in most cross-lingual VC is to separate the speaker and content representation during the training phase and plug back the target speaker and the source content during the decoding stage, similar to speaker disentanglement.

In speaker adaptation, extensive research is being carried out to limit the amount of data required to impart speaker characteristics in the synthesized utterances. Kim *et al.* (2021) uses geometric constraints to learn discriminative speaker representations. A TTS model is trained on a large multispeaker database and fine-tuned using a few minutes of target data. Moss *et al.* (2020) presents the idea of few shot adaptation Bayesian Optimization For FIne-tuning Neural Text To Speech (BOFFIN TTS). Chen *et al.* (2021) developed models to adapt to custom voices and handle different acoustic conditions. Yan *et al.* (2021) developed a TTS model for spontaneous-style speech, which sounds like the target voice. Prakash and Murthy (2020) discusses building generic voices for Indo-Aryan and Dravidian languages, which flexibly scales up to unseen languages and unseen speakers with a few minutes of adaptation data.

2.4 SPEECH SYNTHESIS IN INDIAN CONTEXT : AN OVERVIEW

Speech forms the foundational means of communication between humans. The rapid development in technology has facilitated human-computer interaction through voice in addition to human-human interaction. The process of converting a given text into audio is known as text-to-speech synthesis (TTS). TTS domain has made remarkable progress in recent years. With the advent of deep learning technologies and huge computational resources, TTS has achieved intelligibility and quality equivalent to humans. However, the development in this field was initiated with the fundamental speech synthesis techniques like Unit Selection Synthesis (USS) (Hunt and Black, 1996) and Hidden Markov model speech synthesis system (HTS) (Tokuda *et al.*, 2002). Further, HTS or USS in combination with neural networks was attempted (Ze *et al.*, 2013). Finally, End-to-End (E2E) systems (Wang *et al.*, 2017) came into existence, which is the backbone for the current state-of-the-art text-to-speech synthesis.

The primary speech synthesis systems are - USS, HTS, and E2E. In USS, the segmented and labeled waveforms of the basic units are stored in a database. The labeling is done at the syllable, Akshara, and phone levels. During synthesis, the most similar unit from the database is selected using the target, and the concatenation costs (Hunt and Black, 1996). In HTS, the acoustics and durations are modeled separately. The spectral and excitation parameters are used in acoustics to train context-dependent phone models (Tokuda *et al.*, 2002). This statistical model estimated the parameters based on the data distribution. The explicit acoustic and durational modeling is replaced in the end-to-end framework, which takes text and audio as input and generates speech.

Building speech synthesizers in the Indian context is challenging because Indian languages are digitally low-resourced. India has a wide language diversity with 22 official languages, including English. The lack of accurate transcription and phone-level alignment for each Indian language makes it more difficult to build speech synthesizers. Even within Indian languages, there are two languages families, namely Indo-Aryan and Dravidian. The phonotactics of the two families are different, which has to be taken into account during building synthesizers for Indian languages. A unified approach for parsing text in Indian languages has been proposed by Baby *et al.* (2016*a*).

Semi-automatic and automated techniques for correcting word boundaries have been proposed by Shanmugam (2015). Signal processing cues in tandem with deep learning techniques have been tried by Baby *et al.* (2017) for obtaining accurate phone-level boundaries. Transcription correction for Indian languages using acoustics has been performed by Prakash *et al.* (2018). A unified parser was developed by Baby *et al.* (2016*a*) to parse Indian language text, and a common representation known as Common Label Set (CLS) (Ramani *et al.*, 2013) was developed for Indian languages to handle multi-lingual texts. Thomas *et al.* (2018) attempts to develop code-switched TTS for Indian languages. Further, attempts to build generic voices and adapt to low-resource languages have also been attempted by Prakash and Murthy (2020).

2.5 PARADIGMS USED IN THE THESIS

State-of-the-art voice conversion techniques such CycleGAN (Kaneko and Kameoka, 2018) and StarGAN (Kameoka *et al.*, 2018) were attempted using the conversational speech dataset. However, the results did not seem very promising. To achieve our objective of generating the target speaker's voice in Indian languages in a low resource scenario, we have extended the idea of speaker adaptation proposed by Prakash and Murthy (2020). We have focussed on a unified parser developed by Baby *et al.* (2016*a*) and a common label set proposed by Ramani *et al.* (2013) for handling multilingual text. Since we are dealing with classroom lectures, which are conversational speech, the problem becomes more interesting and challenging.

2.6 SUMMARY

This chapter provided an overview of the literature on conversational speech, voice conversion and adaptation, and Indian language speech synthesis. We also discussed strategies for speech recognition for conversational speech. While ASR has been successful, synthesis of speech with characteristics of conversational speech is difficult. There has been only a little related work in building purely conversation TTS owing to the lack of conversational datasets. Speech synthesis using conversational speech data is indeed challenging. We also discuss the different voice conversion and adaptation

techniques. Further, speech synthesis in the Indian context has been discussed to facilitate the extension of the existing techniques to map English and Indian languages to a common representation.

CHAPTER 3

ANALYSIS OF CONVERSATIONAL SPEECH

3.1 INTRODUCTION

This chapter deals with three aspects — creating a conversational speech dataset, performing a comparative analysis between read speech and conversational speech, and building a conversational speech synthesis model using lecture data. Read speech is the speech recorded by pronouncing a written text directly. It is always semantically correct since it is scripted. The speaking rate of the speaker is maintained more or less constant. It is less expressive. Unlike read speech, conversational speech is spontaneous and prosodically rich (Batliner *et al.*, 1995). It is an impromptu speech and may or may not be semantically correct. Conversational speech captures typical speaker mannerisms, which lead to disfluencies. Due to all these distinctive factors in conversational speech, an in-depth analysis is performed for read speech and conversational speech datasets. First, a conversational speech dataset is created using classroom lecture data from different speakers, as discussed in Section 3.2. This is vital as there is no standard multispeaker conversational speech dataset from classroom lectures. A comparative analysis is performed in Section 3.3 between read speech and conversational speech in terms of syllable rate variation, pitch variation, and Signal-to-Noise Ratio (SNR). Further, typical issues in spontaneous speech such as disfluencies and errors in transcription are highlighted in Section 3.3.5 and 3.4 respectively. An attempt is made to use data pruning techniques to build a robust speech synthesis model using the classroom lecture data in Section 3.5.1. This conversational model is compared with that of a read speech model.

3.2 DATASET DESCRIPTION

Two different datasets have been considered for the analysis and experiments.

3.2.1 Read Speech

For read speech, we have considered a subset of IndicTTS corpus (Baby *et al.*, 2016*b*) comprising 13 Indian languages in Native (vernacular) and Indian English for both male and female voices. Waveforms and the corresponding texts are available. The audio is sampled at 48KHz. The speakers from this dataset are abbreviated as RM1, RF1, and so on, where R indicates read speech, M/F indicates the gender of the speaker and 1 represents the first speaker, and so on. The details of the subset of data used for each task will be discussed in each section.

3.2.2 Conversational Speech

Online educational lectures from National Programme On Technology Enhanced Learning (NPTEL) have been considered for creating a conversational speech dataset. NPTEL is an online educational platform (https://nptel.ac.in/) which has more than 56000 hours of subtitled videos initiated by seven Indian Institutes of Technology and the Indian Institute of Science, Bangalore, in 2003. It is funded by the Ministry of Education (MoE) Government of India. Courses from five Engineering disciplines, namely, civil, computer science, electrical, electronics and communication, and mechanical, as well as Humanities, are offered annually for undergraduate and postgraduate students. Although NPTEL currently offers courses in a few Indian languages, most of the lectures are available in English. Hence we have considered only English videos for our work.

Owning to the lack of open-source conversational speech datasets, we have created our own dataset for this work. Lecture videos are selected from different domains of NPTEL courses. The complete audio of each lecture is extracted from the corresponding video and converted to mono recording from stereo. The audios are passed through a voice activity detector and speech recognizer provided by Speech Lab IITM (https://asr.iitm.ac.in/NPTEL/Transcribe/). This generates a SubRip Text(SRT) file. The SRT file contains the start and end timestamps of speech regions and the corresponding transcriptions generated by ASR. The lecture audios are segmented using the timestamps in the SRT file. With this process, we collected 158.5 hours

of classroom lectures dataset consisting of 21 speakers. The speakers from this dataset are abbreviated as CM1, CF1, and so on, where C indicates conversational, and M and F indicate male and female, respectively. The complete details of this created dataset are given in Appendix A. The dataset will be made available for research purposes on request.

3.3 COMPARATIVE STUDY OF READ SPEECH AND CONVERSATIONAL SPEECH

A comparative study between read-speech denoted as RM1, RM2,....,RF1, RF2,etc. and conversational speech datasets CM1, CM2, ..., CF1, CF2 etc. is performed. Speech parameters such as syllable rate, pitch, and signal-to-noise ratio have been considered for comparison. The details of the dataset used for this analysis task are given in Table 3.1.

Table 3.1: Details of dataset considered for the analysis task

Read Speech			**Conversational Speech**		
Speaker	**Nativity**	**Duration (in hours)**	**Speaker**	**Domain**	**Duration (in hours)**
RM1	Hindi	1/2 (English)	CM1	Computer Science	1/2 (English)
RM2	Kannada	1/2 (English)	CM2	Computer Science	1/2 (English)
RM3	Malayalam	1/2 (English)	CM3	Mathematics	1/2 (English)
RM4	Rajasthani	1/2 (English)	CM4	Physics	1/2 (English)
RM5	Marathi	1/2 (English)	CM5	Humanities	1/2 (English)
RF1	Hindi	1/2 (English)	CF1	Electrical	1/2 (English)
RF2	Tamil	1/2 (English)	CF2	Humanities	1/2 (English)
RF3	Kannada	1/2 (English)	CF3	Humanities	1/2 (English)
RF4	Malayalam	1/2 (English)	CF4	Mechanical	1/2 (English)
RF5	Marathi	1/2 (English)	CF5	Humanities	1/2 (English)

3.3.1 Syllable Rate

Syllables are the fundamental units of speech of the form C*VC*, where V represents a vowel and C* represents optional, one, or more consonants. A syllable is composed of three parts - onset, nucleus, and coda (Bartlett *et al.*, 2009), where the onset and coda correspond to consonants with a vowel at the nucleus. Vowels have long duration and high energy, whereas consonants with low energy form the syllable boundaries. Syllable boundaries can be detected using short-term energy (STE). STE of a speech signal $x[n]$ is computed as :

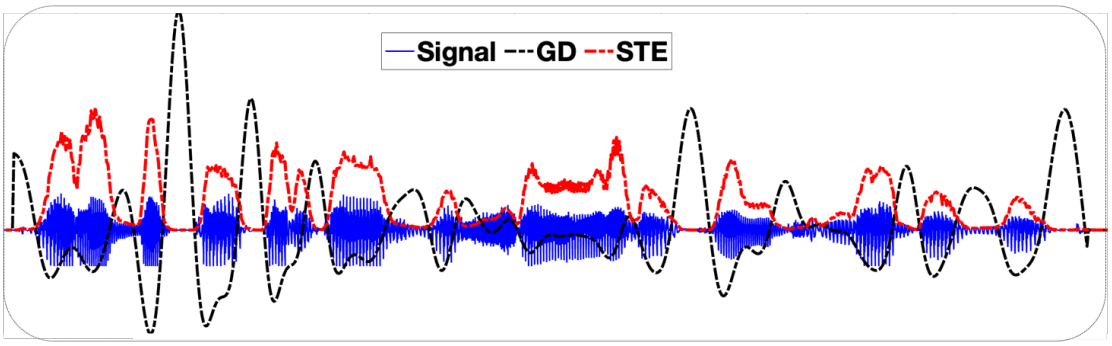


Fig. 3.1: Waveform along with short term energy (STE) and group delay (GD) boundaries

$$E[n] = \sum_{m=0}^{M-1}(x[m].w[n-m])^2$$

where n denotes the current sample, $w[n]$ is the window function and M indicates the frame width.

STE suffers from local energy fluctuations, making syllable boundary detection difficult. Hence, group delay (GD) processing is employed in tandem with STE to obtain a smooth envelope (Prasad *et al.*, 2004). A waveform along with the STE and the GD boundaries are shown in Figure 3.1. The peaks of GD correspond to syllable boundaries. Vowels have a long duration and high energy. Syllable boundaries are obtained in a text agnostic way directly from the audios. Once the boundaries are obtained, the syllable rate is computed. The syllable rate estimates the number of syllables uttered per second, considering only the inter-silence regions. Constant syllable rate across all the utterances is essential for modeling the basic sound units while building text-to-speech synthesis models. In the absence of a constant syllable rate, the quality of text-to-speech(TTS) synthesis output becomes inconsistent.

We manually analyze a few utterances from each dataset to perform a comparative study of the syllable rate of read speech and conversational speech. A waveform along with the spectrogram depicting the syllable boundaries is shown in Figure 3.2 for conversational speech and Figure 3.3 for read speech. Since parallel data is not available across read-speech and conversational speech datasets, different utterances have been considered for the analysis task. The syllables are marked as s1, s2, and so on for each of the utterances. In Figure 3.3 which is an utterance from read-speech, we clearly see that the duration of each syllable s1 to s7 is more or less the same.

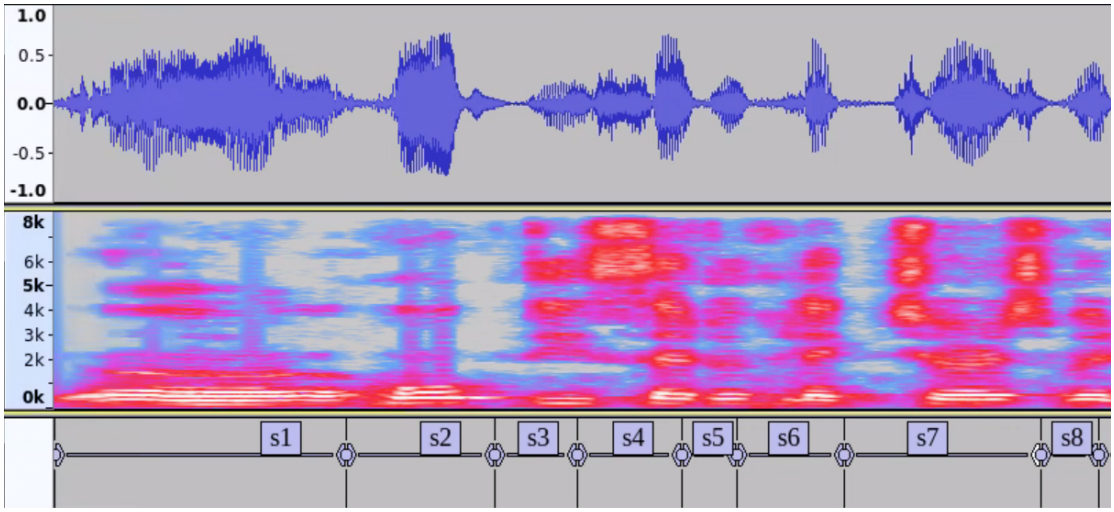


Fig. 3.2: Syllable boundaries for an utterance from conversational speech

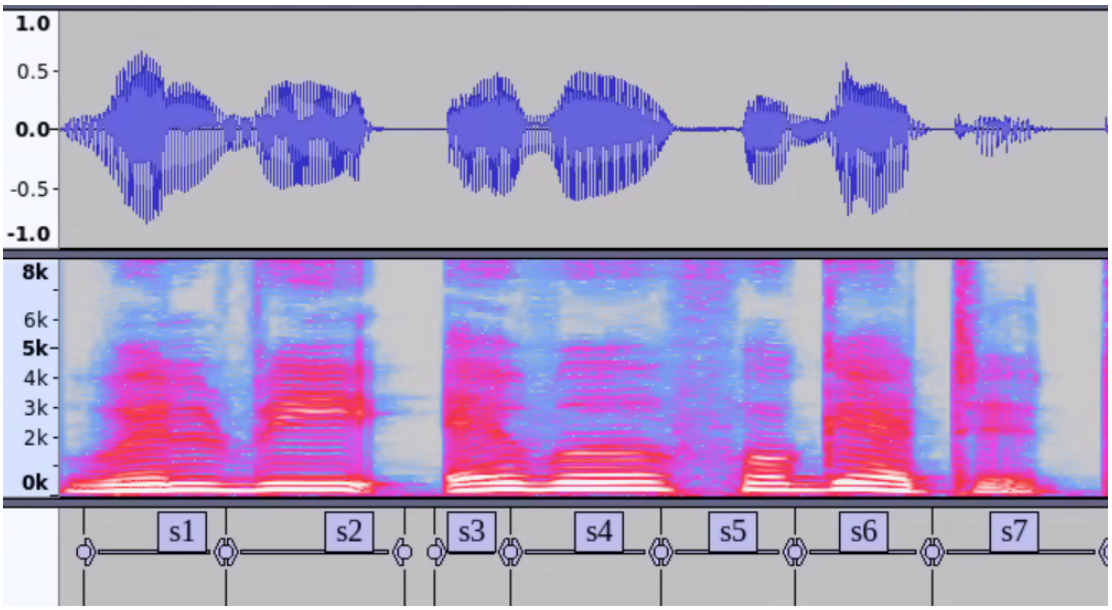


Fig. 3.3: Syllable boundaries for an utterance from read speech

Whereas in Figure 3.2, which is an utterance from conversational speech, the duration of s1 and s7 is quite high compared to the duration of s5 and s8. This indicates the fact that varying syllable rates are an inherent characteristic of conversational speech.

To validate that higher fluctuations in syllable rate occur during conversations, we compute the syllable rates for about 300 utterances of read speech and 300 utterances of conversational speech across different lectures. Figure 3.4 depicts the histograms of syllable rate across multiple utterances of two speakers (RM1, RF1) from read speech and two speakers (CM1, CF1) from conversational speech datasets. As seen from

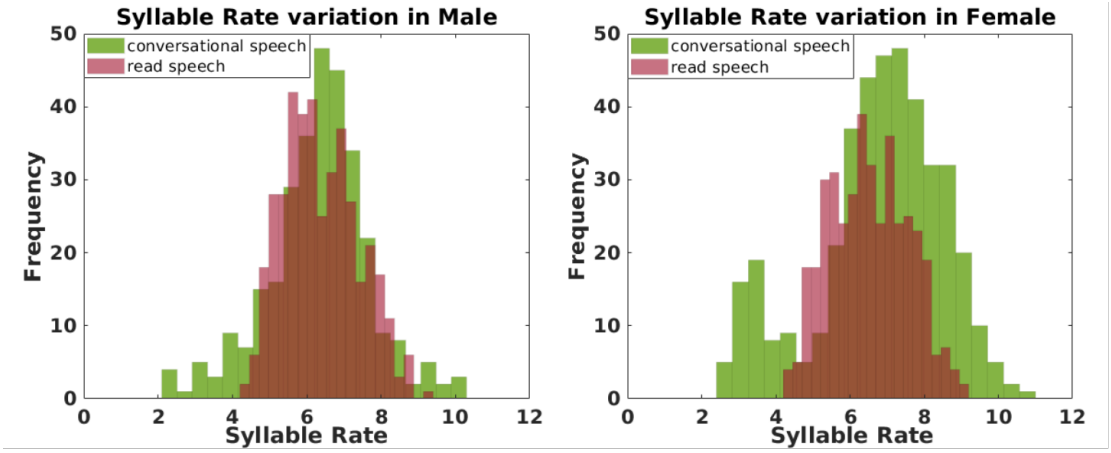


Fig. 3.4: Syllable rate variation between read-speech and conversational speech

Figure 3.4, there is a significant fluctuation in the syllable rate across the utterances obtained from classroom lectures (depicted using green colour). As a result, the variance is higher. In contrast, the syllable rate remains comparatively constant for read speech (depicted using maroon colour). A similar trend is seen in the case of both male and female datasets. Although the two speakers are different and one is read speech and another is conversational speech, there is a certain range of syllable rate common in both speech. The third colour (dark maroon) indicates the overlapping range of syllable rate between the read speech and conversational speech utterances.

The syllable rate is also speaker-dependent to some extent. Hence, to generalize the findings, the syllable rate is computed across ten speakers (5 male, 5 female) each of read speech and conversational speech using the dataset stated in Table 3.1. The average syllable rate and the standard deviation are presented in Table 3.2. It is evident that although the mean syllable rate is almost the same for all the speakers, there is a higher syllable rate variance in the case of conversational speech speakers compared to read-speech speakers.

3.3.2 Pitch

The speech production mechanism can be modeled as a source filter model. The vocal cord vibration generates a train of impulses for voiced signals (Figure 3.5). This forms the source, and the vocal tract shape forms the filter. The fundamental frequency of vibration of the vocal cords is known as pitch.

Table 3.2: Mean and standard deviation(SD) of syllable rate for different speakers

Read Speech			Conversational Speech		
Speaker	**Mean syllable rate**	**SD of syllable rate**	**Speaker**	**Mean syllable rate**	**SD of syllable rate**
RM1	6.38	1.00	**CM1**	6.32	1.36
RM2	6.04	1.04	**CM2**	6.31	1.47
RM3	6.35	1.09	**CM3**	6.15	1.26
RM4	6.29	1.13	**CM4**	6.15	1.35
RM5	6.20	1.13	**CM5**	6.19	1.34
RF1	6.50	1.05	**CF1**	6.79	1.73
RF2	6.73	1.28	**CF2**	6.37	1.32
RF3	6.65	1.04	**CF3**	6.57	1.19
RF4	6.42	1.08	**CF4**	6.57	1.32
RF5	6.46	1.14	**CF5**	6.51	1.38

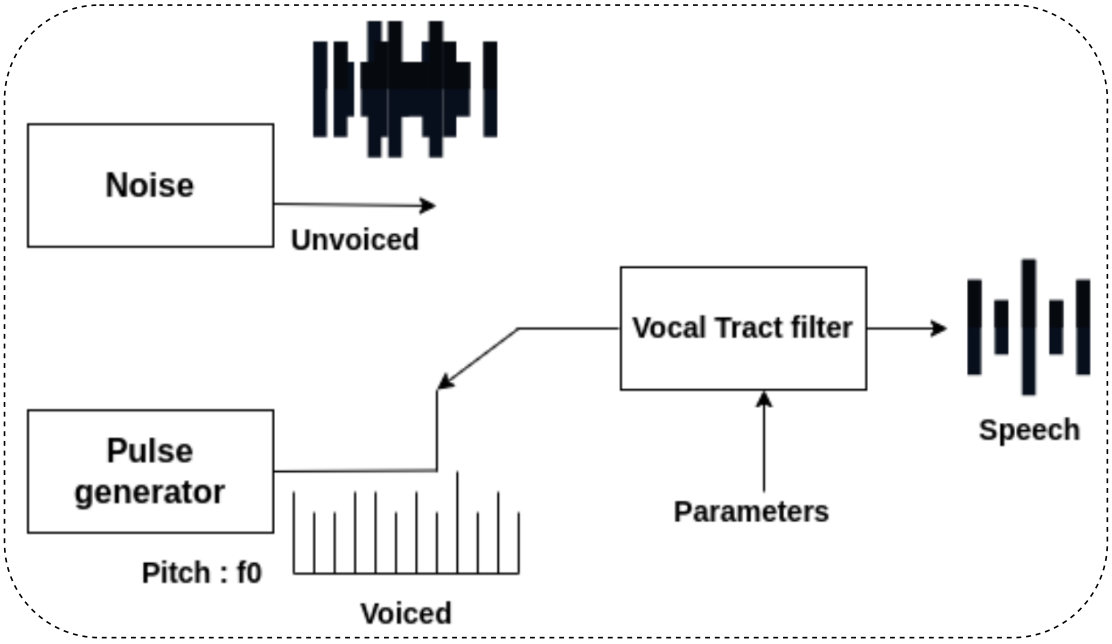


Fig. 3.5: Source-filter model of speech production

Pitch is one of the most critical parameters in speech technology. For any voice manipulation task, modeling the pitch is essential to produce the target voice. Pitch contours help to modify prosodic features and the speaker's intonation. Several analyses have been carried out in the literature for expressive speech (Deo and Deshpande, 2014).

On closely observing the pitch contours of one utterance from each read speech and conversational speech data in Figure 3.6, it is visible that the pitch contours change abruptly even within a single utterance in the case of conversational speech. Even

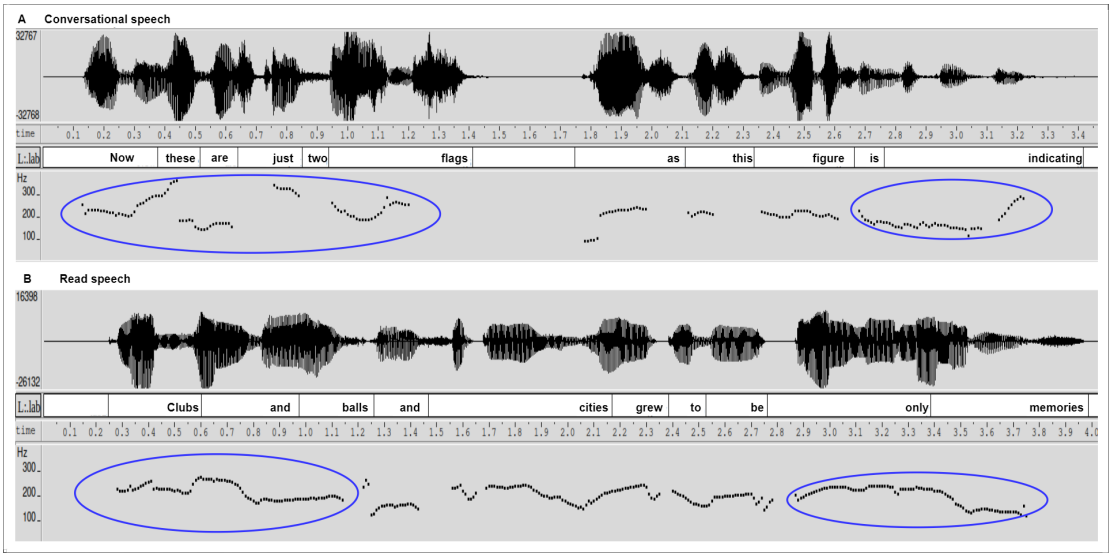


Fig. 3.6: Waveform showing pitch fluctuations in a single utterance of conversational-speech(A) and read speech(B)

at the end of the sentence, where we ideally expect the pitch to become flat, there is a high pitch fluctuation in case A in Figure 3.6. In the second case, i.e., for read speech utterance, the pitch transitions are almost smooth and continuous. The discontinuities and sudden pitch fluctuations in the former result from the expressive nature of conversational speech. Pitch variations are expected in conversational speech to deliver emotions and expressions. During a class lecture, the professor is bound to make prosodic variations to emphasize specific points, which leads to pitch variations. Although prosodic modulations and intonations make conversations more interactive, it adds to the difficulty of building a TTS model using this data. The model struggles to learn the pitch modulations, which may result in pitch variations at random places. Further, we compare the pitch histograms across multiple utterances of read speech and conversational speech of different speakers by plotting the pitch values from the speech regions. From Figure 3.7, it is observed that lecture data (shown in blue) have a wide variation in pitch due to high pitch fluctuations and prosodic variations compared to read speech (shown in orange), even for multiple utterances. The brown region indicates the overlapping range of pitch between read speech and conversational speech utterances. Since the pitch is an essential parameter in speech, the broad pitch range in the case of conversational speech takes a toll on the intelligibility and speaker characteristics of the TTS model. To quantify the findings, the pitch is computed across ten speakers (5

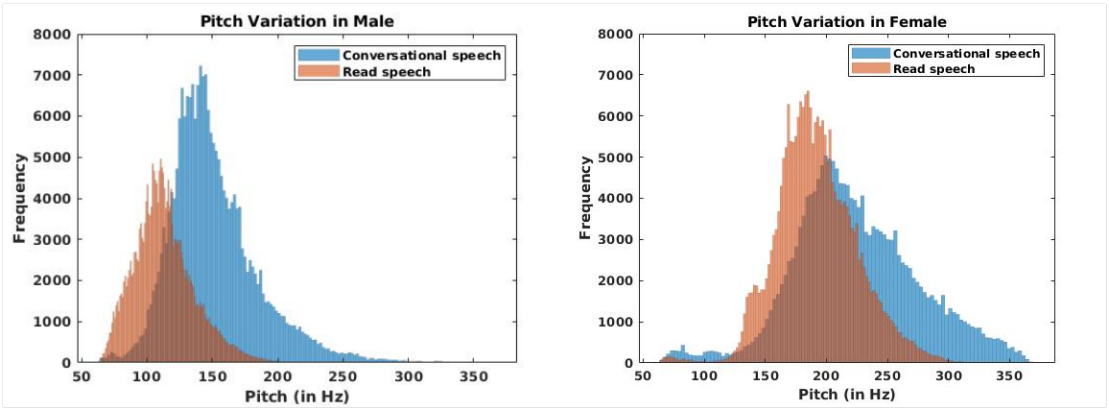


Fig. 3.7: Pitch variation between read-speech and conversational-speech

Table 3.3: Mean and standard deviation (SD) of pitch for different speakers

Read Speech			**Conversational Speech**		
Speaker	**Mean pitch**	**SD of pitch**	**Speaker**	**Mean pitch**	**SD of pitch**
RM1	114.78	24.30	**CM1**	159.42	37.19
RM2	119.81	35.55	**CM2**	149.22	44.28
RM3	134.39	20.90	**CM3**	154.10	38.19
RM4	113.37	24.46	**CM4**	144.03	45.45
RM5	117.55	19.55	**CM5**	159.03	47.15
RF1	191.23	34.05	**CF1**	219.18	48.92
RF2	204.03	44.33	**CF2**	222.82	52.52
RF3	230.14	56.20	**CF3**	181.82	41.18
RF4	216.16	50.13	**CF4**	246.10	46.61
RF5	231.94	46.91	**CF5**	201.23	53.92

male, 5 female), each of read speech and conversational speech. The average pitch and the standard deviation are presented in Table 3.3. For the majority of the cases, the trend, i.e., high variance of pitch for conversational speech speakers, is observed when compared to read speech speakers.

3.3.3 Signal to Noise Ratio

Evolving from the fundamentals of digital signal processing, speech is a form of signal. The quality of the signal is highly influenced by the recording conditions, background noise, recording devices, etc. A higher SNR ratio indicates that the signal power is more, whereas a lower SNR ratio indicates that the noise is more. As mentioned in Section 3.2, the read speech data has been recorded in a studio environment and

carefully curated. Although the conversational speech data from NPTEL are also studio-recorded, other factors like the microphone used during recording, background noise, classroom noise, etc., come into consideration. SNR is the ratio between the signal power and the noise power. SNR is usually expressed in decibel (dB) and is computed as :

$$SNR_{dB} = 10log_{10}\frac{P(signal)}{P(noise)}$$

In Figure 3.8, we plot the spectrograms of one single utterance of read speech and one utterance of conversational speech. The lecture data in Figure 3.8 has more high-frequency components, indicating noise compared to the read speech utterance.

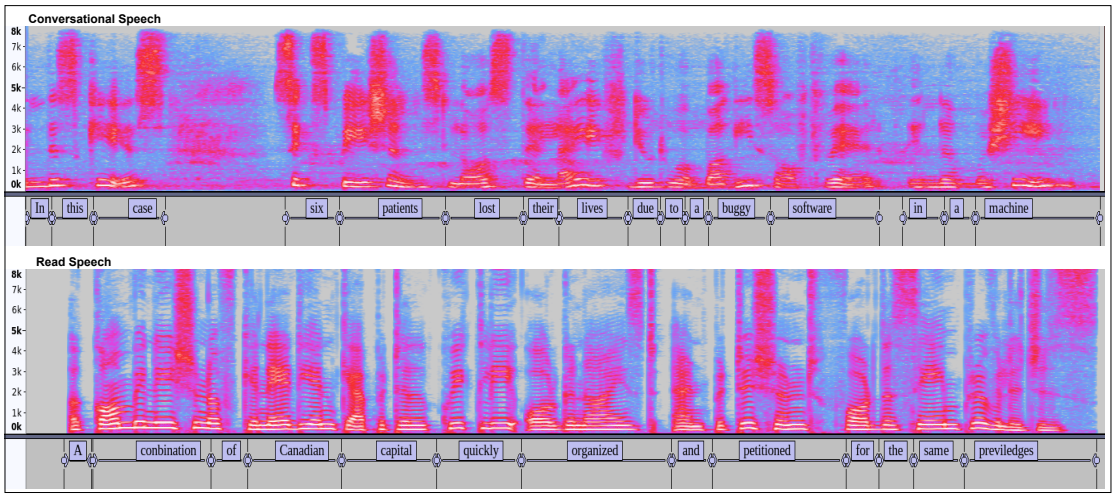


Fig. 3.8: Spectrogram comparing one utterance of conversational speech (up) read speech(down)

Further, we compute the SNR values across ten speakers (5 male, 5 female) each of read speech and conversational speech. The SNR is computed using SNR estimation on Waveform Amplitude Distribution Analysis (WADA) as stated by Kim and Stern (2008). It is assumed that the noise and speech signals are not intertwined. Clean speech forms a gamma distribution, whereas the background noise composes a Gaussian distribution. Based on this assumption, the SNR values are computed. An SNR value greater than 20dB is considered to be clean speech; hence we compute the percentage of data that has an SNR value less than 20dB. The average SNR is also shown in Table 3.4. It is clearly evident that for all read speech speakers, the SNR value is very high, and hence the data is very clean. In contrast to this, the conversational

speech data has some percentage of noisy utterances even though the mean SNR appears to be greater than 20dB. These noisy utterances may account for degradation in the TTS quality if this data is used for building speech synthesizers.

Table 3.4: SNR values for different speakers

Read Speech			**Conversational Speech**		
Speakers	**Mean SNR**	**% less than 20 dB**	**Speaker**	**Mean SNR**	**% less than 20 dB**
RM1	100	0	**CM1**	73.16	4.31
RM2	100	0	**CM2**	44.71	4.19
RM3	100	0	**CM3**	68.72	6.08
RM4	100	0	**CM4**	32.04	33.7
RM5	97.77	0.51	**CM5**	39.14	10.89
RF1	100	0	**CF1**	42.36	0.28
RF2	84.89	0	**CF2**	40.32	11.48
RF3	100	0	**CF3**	70.79	0
RF4	100	0	**CF4**	32.48	6.68
RF5	100	0	**CF5**	29.76	11.43

3.3.4 Feature Space for Read Speech and Conversational Speech

As evident from the above discussions, we understand that pitch, syllable rate, and SNR are essential parameters for distinguishing between read speech and conversational speech. Here, we attempt to find a feature space where the two (read and conversational) groups of speakers are distinctly separated. We have considered the dataset as given in Table 3.1 for this task. We observe that the pitch deviation and the syllable rate deviation are higher for conversational speech; hence we consider the deviation instead of the mean values. We plot syllable rate deviation vs. SNR in Figure 3.9, pitch variation vs. SNR in Figure 3.10, and syllable rate deviation vs. pitch variation in Figure 3.11 for ten speakers. The blue colour is used to represent read speech speakers, and the red colour is used to represent conversational speech speakers. To distinguish between genders, an inverted triangle is used for males, and a circle is used for females. Read speech and conversational speech speakers are well-separated in a space formed by syllable rate deviation vs. SNR, as shown in Figure 3.9. SNR and syllable rate deviation are uncorrelated features. The blue points (read speech) have a lower standard deviation

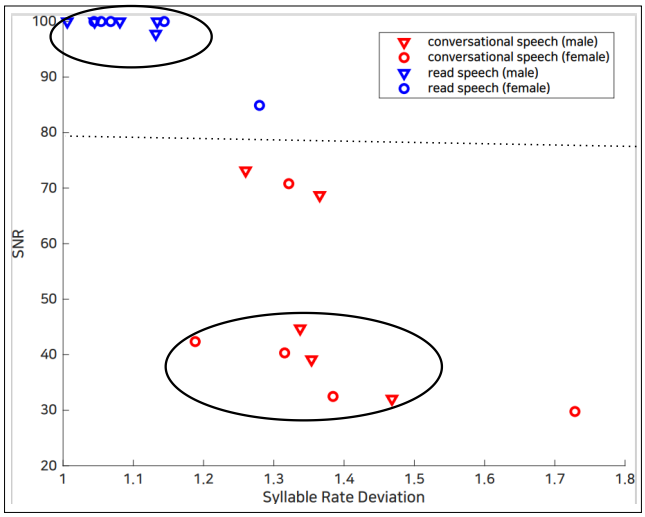


Fig. 3.9: Speakers' representation in syllable rate deviation vs SNR space

of syllable rate and a higher SNR value, clearly forming a distinct cluster. Similarly, the conversational speech speakers create another distinct cluster almost closely spaced, and a linear decision boundary can be observed separating the two groups of speakers. Even though the gender is different, we observe that read speech speakers are very closely spaced, indicating the underlying similarities in their attributes.

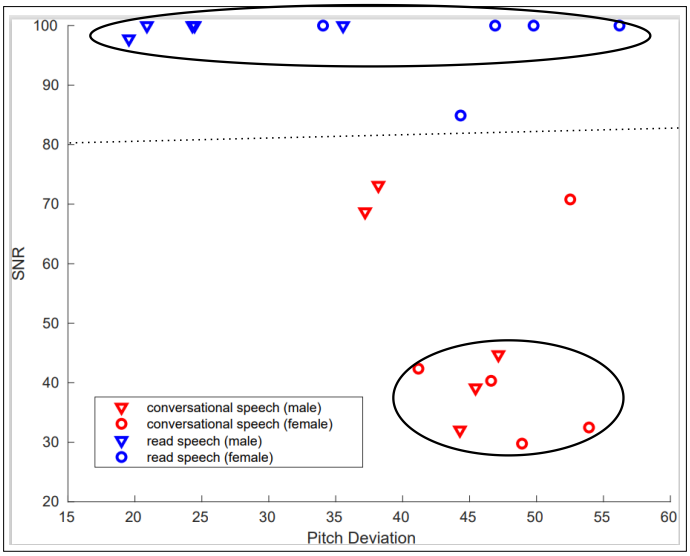


Fig. 3.10: SNR of Speakers' and corresponding pitch variation

In Figure 3.10, the read and conversational speakers form separate clusters even considering pitch variation vs. SNR. The blue points (read speech) have a higher value of SNR. It is noteworthy that the pitch variation for male data is less compared to female data for both read and conversational speech, as seen in the plots. However,

generalization in pitch deviation for read and conversational speech cannot be made distinctly because it also depends on other factors like voice timbre and prosody of the particular speaker. This could also be because of recording environments or errors in the pitch extraction algorithm. A linear decision boundary can still be considered to separate the speakers of these two groups.

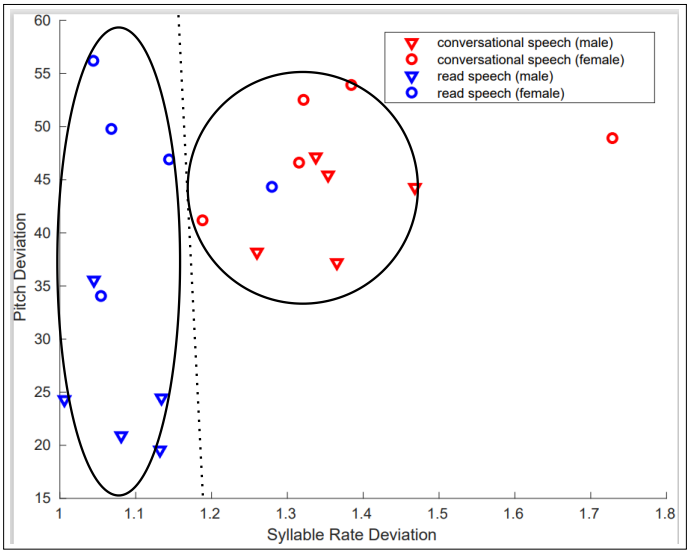


Fig. 3.11: Syllable rate deviation vs. pitch variation

Similarly, in Figure 3.11, we observe two clusters - one for read and one for conversational speakers. The syllable rate deviation mainly impacts the separation, and the decision boundary segregates the two into different groups with just one outlier. This demonstrates that considering the uncorrelated feature spaces already discussed, we can classify speech as read speech and conversational speech.

3.3.5 Disfluencies

Disfluencies are unavoidable characteristics in spontaneous speech. Classroom lectures are unscripted; hence lots of disfluencies like *umm*, *ah*, *okay*, *so*, *right*, *is it*, etc., such filler words occur during a lecture. False starts can also occur while delivering a lecture, along with these filler words. This makes it difficult for the ASR language model to make correct transcript predictions. The language model is mostly trained on clean text, not conversational. So disfluency detection, as well as removal, becomes challenging. The error in transcription prediction may lead to a mismatch between

Table 3.5: Percentage of disfluencies in 30 mins lecture audio (conversational speech)

Speaker	No. of disfluencies	Total no. of words	Disfluency rate
CF1	194	4750	4.08%
CM1	162	4920	3.29%

the audio and corresponding transcriptions. Automatic disfluency detection is more difficult because certain words like *okay*, *so*, *right*, etc., can occur even as a part of the text. Figure 3.12 illustrates an example of a speech segment with the disfluencies *Ok* and *Ah* captured during a lecture. It is noteworthy that the duration of *Ah*, being a filler word, is very long. As a result, the syllable rate of utterances having such filler words is less since the disfluent vowel is elongated. This in turn might affect the performance of the TTS model trained using this data. Due to the manual effort involved in disfluency

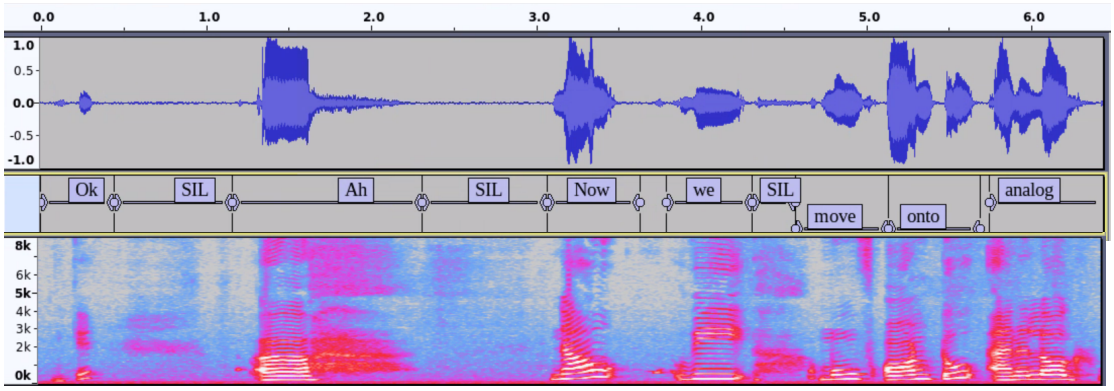


Fig. 3.12: Disfluencies *"Ok"* and *"Ah"* in a lecture segment

identification, analysis is performed on one male (CM1) and one female (CF1) lecturer for conversational speech. About 30 minutes of the course is considered for analysis. Disfluency rate is defined as the ratio between the number of disfluent words to the total number of words. As seen from Table 3.5, the average disfluency rate of conversational speakers is about 4%. There are no disfluencies in read speech because the recordings are rehearsed and scripted.

3.4 ISSUES IN ASR

One of the major challenges in conversational speech is the lack of transcriptions for the audio. Getting accurate manual transcription is costly. So we depend on ASR models to generate the text. The current state-of-the-art ASR models, which have already

achieved a very low word error rate for read speech (Baevski *et al.*, 2020), struggle when the data is conversational. For our work, an ASR trained in technical domains is used for transcribing lecture videos. The issues in the ASR prediction result in a mismatch in the audio-transcription pairs, thereby making the TTS training furthermore difficult. Figure 3.13 presents the most commonly occurring errors in ASR. The system-generated outputs are marked in red, and the expected transcriptions are marked in blue. The most commonly encountered issue during manual correction of transcription

Type of Issues	ASR Output	Correct Transcription
Type 1: Erroneous ASR	ASR: Why fiber optics is the **backward** for all communication systems?	Actual: Why fiber optics is the **backbone** for all communication systems?
Type 2: Similar sounding words	Now, think about your home **your** watching television in home you have	Now, think about your home **you're** watching television in home you have
Type 3: Disfluencies	at every home every hostel where **you are you are** connected through a high speed communication line right.	at every home every hostel where **you are** connected through a high speed communication line right.
Type 4: Proper Nouns	And the good reference is the introduction to fiber optics book by professors **gata and kagaragin ah**	And the good reference is the introduction to fiber optics book by professors **Ghatak and Thyagarajan**
Type 5: Equations, Mathematical symbols, abbreviations, numbers	all the homes are connected through what is called as **ft. the h** links.	all the homes are connected through what is called as **FTTH** links.

Fig. 3.13: Types of issues encountered in ASR transcriptions

is due to Type 1 (Figure 3.13). These errors can occur due to mispronunciation and accents typical to a particular speaker. Type 4 is due to Out-of-Vocabulary (OOV) words since the proper nouns may not be seen during the ASR training phase. Added to this, domain-specific technical terms might occur during the decoding time. As a result, the ASR predicts a similar word as seen during training. A key point to note here is that these technical lectures are conversational in nature and therefore do not follow sentence structures and have disfluencies. These result in issues of Types 2 and 3, as shown in Figure 3.13 which occur due to false starts. Technical lectures are bound to have a

lot of mathematical symbols, equations, and abbreviations. Handling the symbols and equations consistently throughout the lecture becomes difficult for the ASR. As shown in the Type 5 issue, 'T' is getting confused with 'the'. Hence it is very crucial to devise techniques to prune utterances that exactly match the transcription before training any TTS model to reduce this mismatch.

3.5 TTS USING CONVERSATIONAL SPEECH : CLASSROOM LECTURES

After understanding the fundamental differences between read speech and conversational speech, we attempt to build a text-to-speech synthesis model using the lecture recordings and compare it with that of a read speech model. The details of data used for training the read speech and conversational speech models are given in Table 3.6.

Table 3.6: Details of data used for building character based end-to-end systems

Read Speech			**Conversational Speech**		
Speaker	**Nativity**	**Duration (in Hrs)**	**Speaker**	**Domain**	**Duration (in Hrs)**
RF6	Bengali	8.5 (English)	CF6	Computer Science	39.6 (English)

We understand that the task of building a conversational TTS is very challenging as there are multiple issues. The audios have variable syllable rate, pitch, and SNR, as discussed in Section 3.3. Further, the <*text, audio*> pairs are not 100% accurate since they are machine-generated and not manually curated. Also, there is a lack of any other curated conversational data which can be leveraged to bootstrap the TTS model. To overcome the shortcomings in conversational speech, we propose a pruning module and attempt to bring the conversational data close to that of read speech data.

3.5.1 Pruning Module

Figure 3.14 shows an overview of the pruning module. The audios are pruned using parameters such as syllable rate and alignments, and denoised. The set of curated audios is used for building an E2E conversational TTS model. For syllable rate and SNR computation, only audios are needed, whereas for alignments, the audios, as well as the corresponding text, are needed. Pitch has not been considered as a parameter for

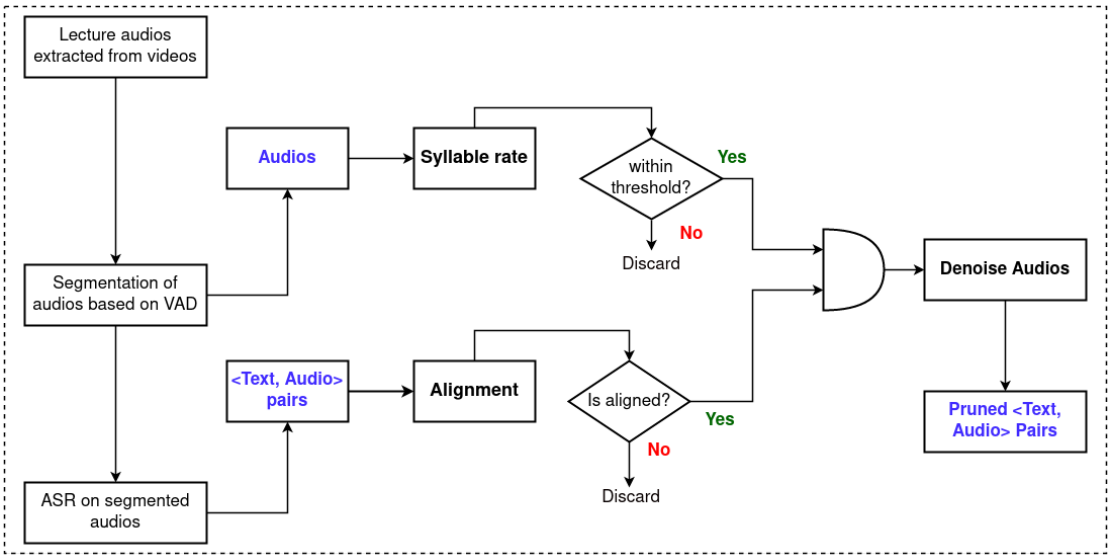


Fig. 3.14: Block diagram of pruning module

pruning because the prosody and the expressive nature, in the case of conversational speech, are due to these pitch transitions. As our objective is to build models which are more natural, we do not discard utterances with high pitch variation. Here, we have used the data of the speaker represented as CF6 in Table 3.6. Each of the pruning steps is discussed below:

Syllable-rate

In Section 3.3.1, we discussed the method of computing the syllable rate of each utterance. The mean and SD of the syllable rate of all the utterances are computed. The mean is found to be 7.17, and the SD is found to be 2.41. To avoid utterances having very high or very low syllable rates, we set a threshold of $\pm$ 1SD, and we discard the utterances which do not lie within the threshold. The new mean and SD are found to be 6.91 and 1.46, respectively. The syllable rate distribution before and after pruning is shown in Figure 3.15. Since disfluent utterances might result in a low syllable rate, the assumption is that the disfluent utterances also get pruned during this stage.

Alignments

As discussed in Section 3.2, the transcriptions of the lectures are obtained from ASR models. Hence, they might have insertion, deletion, or substitution errors. So, a word-

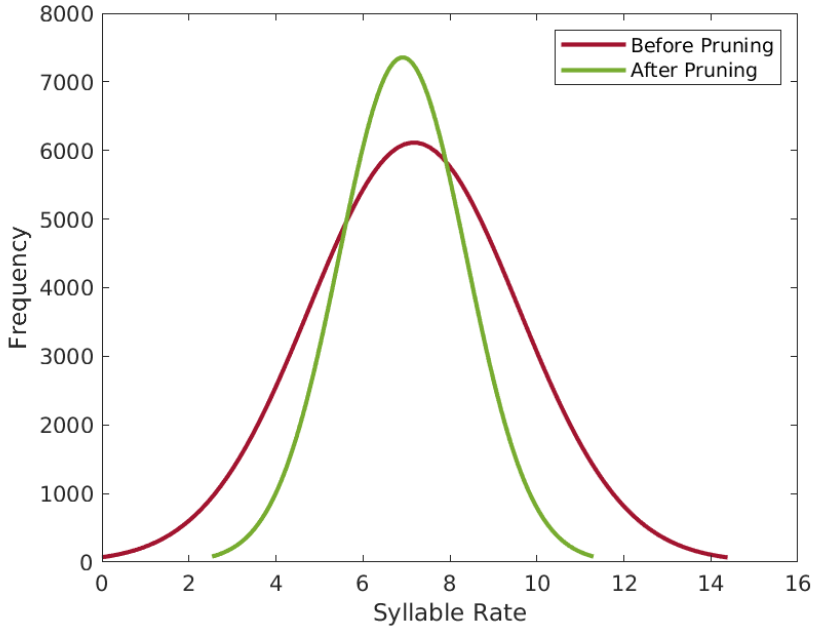


Fig. 3.15: Syllable rate distribution modelled as a Gaussian before and after pruning

level alignment check is performed as a part of the pruning stages to ensure that the utterances to be used for building the conversational TTS model are error-free as far as possible. Kaldi framework (Povey *et al.*, 2011) has been used to check the word level alignments of the lecture data. A monophonic model is trained using about 50 hours of the conversational speech data of multiple speakers combined together, as stated in Table 3.1 and Table 3.6. The high amount of training data compensates for the minor mismatch in the audio-transcription pairs, and the monophonic models become robust. The training data of speaker CF6 (Table 3.6) is further decoded back using these monophonic models. It is noticed that for a few utterances, the word-level alignments are not generated. This is because if a word is missing either in the audio or in the transcription, the text-audio pair cannot be aligned. This indicates the mismatch between the audio transcription pairs. The utterances for which the alignments are not obtained are discarded at this stage. Additional data of different speakers have been seen to improve the performance of the alignment accuracies. This alignment step ensures that utterances with word skips or undetected disfluencies are discarded.

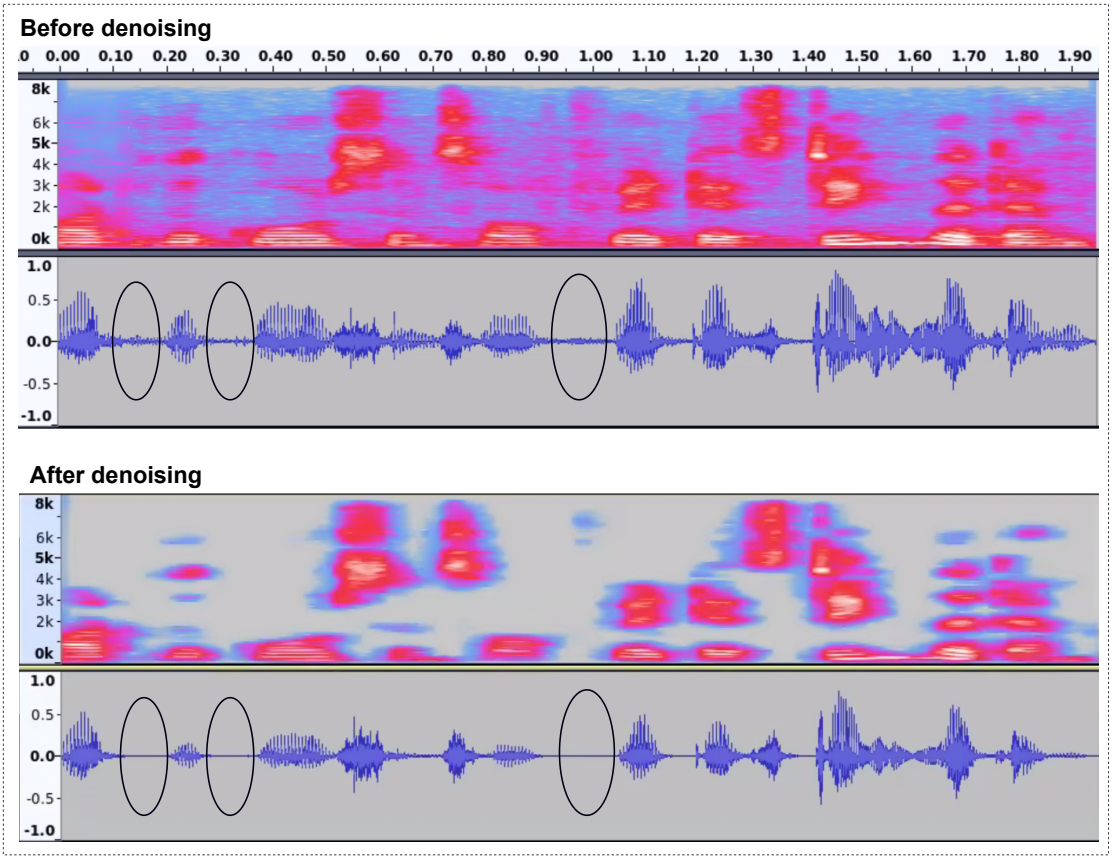


Fig. 3.16: Spectrogram and waveform showing an audio before and after denoising

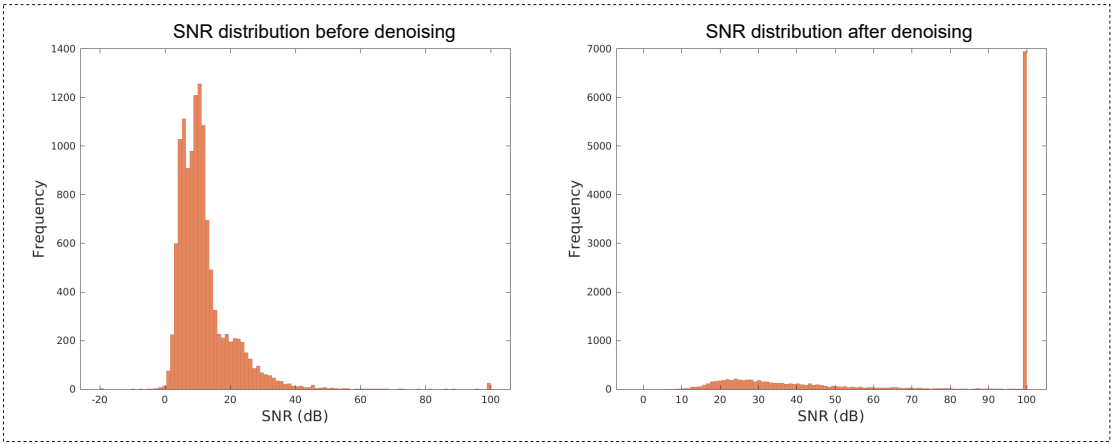


Fig. 3.17: SNR distribution before and after pruning for one speaker

Denoising

As discussed previously, the recorded lecture data have some noise. The SNR for the data is computed using WADA SNR (Kim and Stern, 2008) as stated in Section 3.3.3. In Figure 3.17, we observe that majority of the utterances lie in the range of SNR less

than 20dB. On discarding these utterances, we would not be left with enough data for building a TTS model. Hence, we attempt to denoise the audio using the *noisereduce PyPi package* (Sainburg, 2019; Sainburg *et al.*, 2020). The package uses the spectral-gating method to estimate the noise threshold from the spectrogram of the audio and denoises the audio.

Figure 3.16 shows one utterance before and after denoising. We observe that we lose out on some information in this process of noise removal. But still, the silence regions are captured in a better way, as marked in Figure 3.16. In the presence of noise, we were unable to train a TTS model for conversational speech. The SNR distribution before and after noise removal for speaker CF6 is shown in Figure 3.17.

3.5.2 System Building

The TTS systems discussed further are trained using End-to-End architecture. ESPnet's implementation (Watanabe *et al.*, 2018) of Tacotron2 (Shen *et al.*, 2018) is used for this task. Tacotron2 employs an attention mechanism in its encoder-decoder design. The audios greater than 15 seconds are discarded to help the model learn the attention more robustly. For training, just text-audio pairs are required. The set of English characters, along with a <unk> token for unknown characters, <space> token for space, <sos/eos> token for sentence beginning or end, and <blank> token for blanks, comprises the dictionary. A character-based model was trained using a dictionary size of 30. This set of characters is mapped into distinct tokens. The encoder model receives these tokens and converts them into fixed-length vectors, which are also known as character embeddings. The decoder receives the character embeddings and predicts the Mel-spectrogram for each frame. The Tacotron-2 architecture takes the text and the audio spectrograms as input. Raw audio spectrograms are used as input since they provide rich information about the acoustics and the intensity of the spoken utterances. Spectrograms are considered over a short window since speech signals are assumed to be quasi-stationary. The basic overview of training and synthesis using an End-to-End framework is shown in Figure 3.18. The raw spectrograms are converted back into audio waveforms using a vocoder. Waveglow (Prenger *et al.*, 2019) is the vocoder in our case. It takes mel-spectrograms as input and generates back the audio. Only the

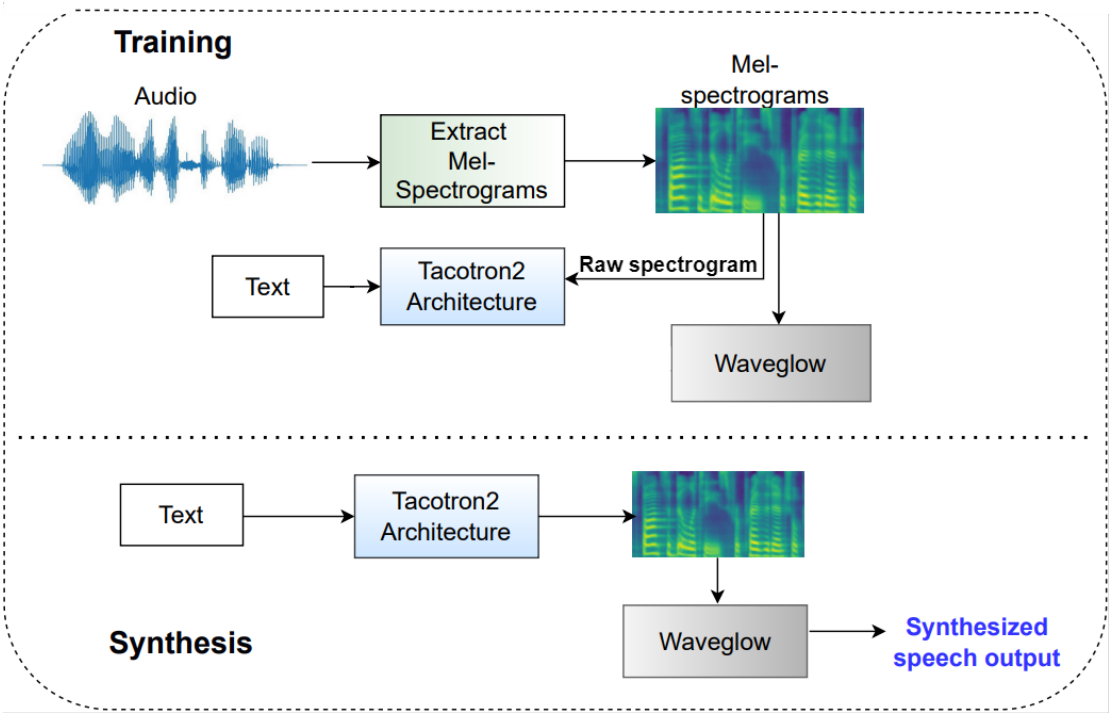


Fig. 3.18: Block diagram of End-to-End training and synthesis modules

mel-spectrograms extracted from raw audios are needed for the training of waveglow. The pre-trained LJSpeech (Ito and Johnson, 2017) waveglow model is used as an initial model and finetuned using the speaker's data. The synthesis pipeline has two tasks : Spectrogram generation and vocoder. We do not make any modification to the vocoder during synthesis.

To perform a comparative study between read speech and conversational speech TTS, two models are trained. The read speech model (System 2) is trained using English data of speaker RF6 as given in Table 3.6. An attempt was made to train a conversational speech model by pooling lecture data of CF6 as stated in Table 3.6 without pruning. However, the model was not intelligible at all. Hence, the pruning module was introduced to make the conversational data (lecture data) similar to read speech data to be used for building TTS. The <text, audio> pairs obtained from the pruning module are assumed to be suitable for building a conversational TTS model. This data is from the same conversational speaker, CF6. Data of about 18 hours obtained after pruning was used to train System 1. The block diagrams of the two systems trained in this section are shown in Figure 3.19. As seen from the figure, System1 (conversational

TTS) has the pruning module because the data is from classroom lectures which are essentially conversational. System2 (Read speech) does not need the pruning module since it is read speech data that is rehearsed/ read out at a constant speaking rate.

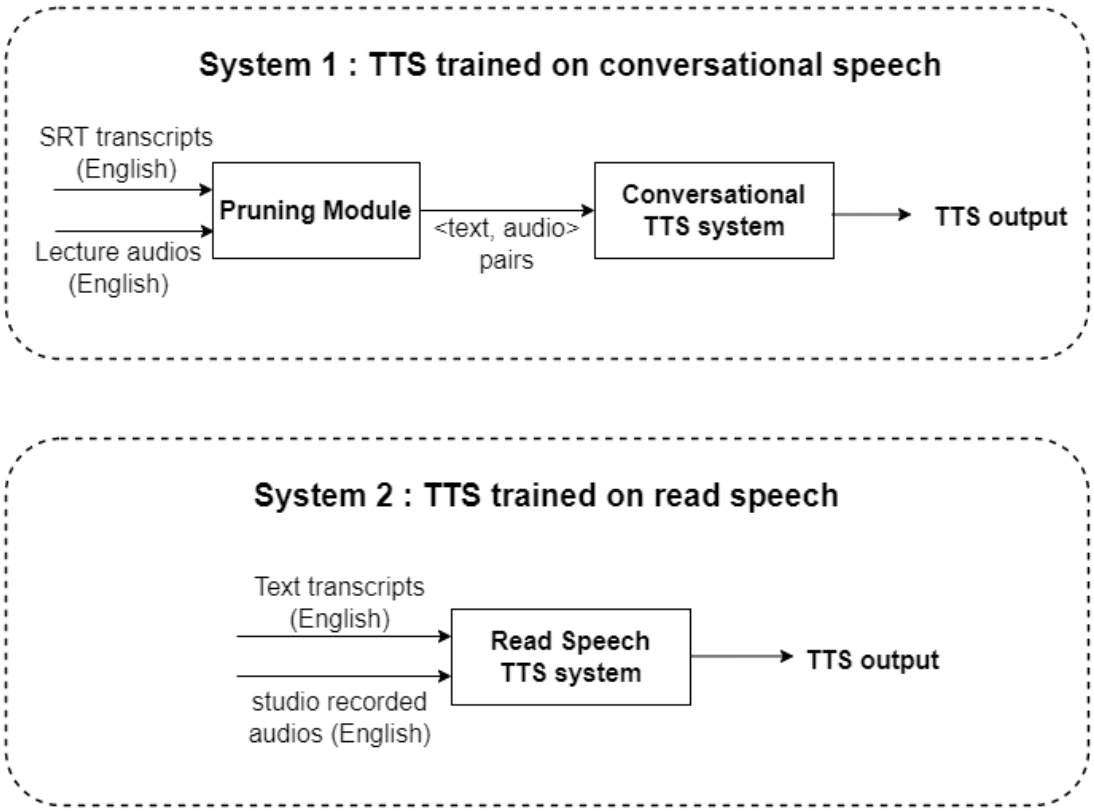


Fig. 3.19: TTS Systems

Evaluation

After introducing the pruning module, the conversational TTS System1 could generate a few intelligible utterances. However, still, the model was not equivalent to read speech TTS and had issues with respect to word clarity and mistakes. Out of a total of 1635 unseen test sentences, 23 utterances were not very intelligible. For evaluation of the conversational TTS system (System 1), pairwise comparison (PC) test and degraded mean opinion score (DMOS) (Viswanathan and Viswanathan, 2005) test have been performed. Only intelligible audios from System1 were evaluated and compared with System2. Since ASR errors like insertions, deletions, and substitutions still persist in System 1, we further performed a word error rate (WER) computation. The sample audios are present in the link (https://www.iitm.ac.in/donlab/preview/test/index.html)

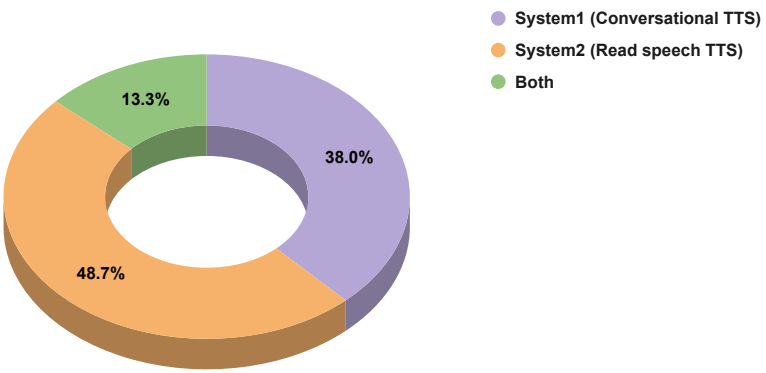


Fig. 3.20: Result of pairwise-comparison test for System1 and System2

PC test: In the Pairwise Comparison test, two audios (one from each system) are played to the evaluators in each trial. The order of the systems is shuffled in each test case to prevent bias. The evaluators are asked to rate their preference for the audio based on intelligibility and naturalness. It was very difficult for the evaluators to perceive naturalness and intelligibility separately and give a rating; hence both of the tests were combined together for evaluation. The ratings are performed on a scale of 1-5 (5 being the best). If both audios are preferred equally, an equal rating is given. PC test is performed here to identify the preference of listeners between TTSes trained on read speech and conversational speech. Ten unseen test transcriptions from lectures have been chosen and synthesized using both systems. Fifteen evaluators participated in the test. Figure 3.20 shows the results of the PC test. It is seen that System 1, i.e., the conversational TTS, was preferred 38% of the times while System 2, i.e., the read speech system, was preferred 48.7% of the times and 13.3% times both the systems were rated equally. Informal evaluations suggested that multiple factors influenced a person's preference. Since both the speakers are different, the preference of one over the other could be due to the voice timbre, intelligibility, or naturalness. In a few cases, the naturalness was more in the case of System 1, whereas intelligibility was more in System 2. System 1 sounded more natural because of the typical speaker mannerisms and prosodic variations, which were not present in the case of read speech. However, it took a toll on the system intelligibility in a few cases.

DMOS test: Degraded Mean Opinion Score (DMOS) (Viswanathan and Viswanathan, 2005) is a popular technique for evaluating speech synthesis quality. In the DMOS test, the synthesized utterances from each system are presented to each listener along with a few natural sentences in random order. Each listener is asked to rate speech quality based on naturalness and intelligibility, with 5 as the most natural-sounding utterance and 1 as the least. A precision of 0.5 is allowed, and the framework is set such that no rating is allowed until the entire utterance is played. The score for each system is normalized with respect to the score obtained by the natural sentences. The DMOS for each system for one listener is computed as :

$$DMOS(System) = \frac{AverageScore(System)}{AverageScore(Natural)}$$

Finally, the DMOS score of each system is computed by taking the mean rating given by all the listeners for that particular system.

The read speech TTS (System2) has only been included for the PC test since DMOS for read speech has already achieved quality equivalent to natural speech. For System 1, the DMOS score is given in Table 3.7. Although the DMOS score is less than that of read speech TTS systems, the results are still encouraging as it shows that building a completely conversational system using classroom lectures is possible using data pruning techniques. The results can be improved by identifying more sophisticated techniques for data curation as well as improved ASR.

Table 3.7: DMOS score for conversational TTS

System	DMOS Score
System1 (Conversational TTS)	**3.37**

WER test: An arbitrary set of 20 sentences synthesized using System 1 was informally evaluated to find the number of insertion, deletion, and substitution errors, and the result is shown in Table 3.8. The Word Error rate (WER) test was not performed for read speech because the WER was close to 0. We observe that conversational speech TTS has word skips, substitutions, and insertions even after pruning the data used for training.

The reason might be due to the unstructured and incomplete sentences seen during training of the system.

Table 3.8: Number of insertions, deletions and substitutions

Total no. of words	No. of insertions	No. of deletions	No. of substitutions
318	4	38	16

3.6 SUMMARY

This chapter overviewed the underlying differences between read-out transcriptions and spontaneous lectures. The factors like syllable rate variation, pitch variations, signal to noise ratio, which are responsible for differentiating conversational speech from read-speech, were identified, and a comparative study was made between different speakers from each of the two groups. Such an exhaustive study is critical to analyze, understand and give future research direction using spontaneous speech.

Factors like disfluencies and ASR transcription errors were indicated, adding to the difficulty due to the lack of manually curated transcripts. Due to the enormous manual effort involved in curating the lecture data, a data engineering approach of pruning was identified and applied. A preliminary attempt was made to build TTS models using this pruned classroom lecture data, which is an extremely challenging task and first of its kind. Evaluations and comparisons with TTS trained on read speech were performed. The results indicate that conversational data preserve the naturalness and sound more interactive compared to read speech TTS. However, the model output is not very intelligible in some instances due to the variability in context, as the training sentences may not be complete and meaningful. There are scopes of improvement in conversational TTS; it is not at par with read speech systems in terms of intelligibility. But, the analysis and results are encouraging as it paves the way for exploration towards conversational TTS, voice adaptation, speaker identification, and much more using conversational data.

CHAPTER 4

CROSS-LINGUAL SPEAKER ADAPTATION

4.1 INTRODUCTION

Chapter 3 discussed the challenges in building a text-to-speech (TTS) synthesis model using conversational classroom lecture data. The conversational speech system is trained using the lecturer's data using data pruning techniques. However, our ultimate objective is to generate the professor's voice in a target Indian language. The conversational system discussed in Chapter 3 has the speaker's voice characteristics but is trained only in English; hence cannot synthesize text in any other Indian language. Another obstacle is that we do not have any Indian language conversational speech dataset, which limits exploration in this field. Added to the data sparse situation, the data is conversational, making it very difficult to build robust TTS models. To overcome the data limitations, we propose to adapt read speech models using a minimum amount of conversational speech to generate the target speaker's voice in a target language.

The proposed approach follows the work by Prakash and Murthy (2020) where a generic TTS model is built by pooling multiple Indian languages and adapted to a target language. A generic model can capture a wide range of acoustics, can be scaled up in a low resource scenario, and is computationally inexpensive. The paper states that target speaker characteristics are preserved even with as less as seven minutes of adaptation data. The work focuses on building generic TTSes for Indo-Aryan (Hindi, Bengali, Rajasthani, etc.) and Dravidian (Tamil, Kannada, Telugu, etc.) languages separately as the language families have distinguishing characteristics. However, the work is primarily done on read speech and Indian languages. Adaptation between Indian languages and English, which are phonotactically very different, has not been attempted previously. Further, using conversational speech to adapt read speech systems within languages with a significant difference in phonotactics to impart speaker characteristics is a more ambitious objective.

The idea is to use read speech to build a robust base model in an Indian language. To account for the phonotactic variations between English and Indian languages, an

equal amount of English data of the same read speech speaker is used, and a bilingual (English + Indian language) model is trained. Using read speech data for the base model is inspired by the fact that read speech data is present for multiple Indian languages as opposed to conversational data, which is present only in English. Hence, the objective of speaker adaptation can be scaled across various Indian languages. There is no mismatch in the audio-transcript correspondence in read speech, so the base model can be trained without any issue. In the case of conversational speech, the transcripts are ASR generated, which may not be completely error-free. Read speech has a constant syllable rate and pitch, and no disfluencies, which helps build robust TTS models. The base model captures the phonotactics and linguistics of the Indian language, which is adapted to a different speaker using only a small amount of English data to generate the speaker characteristics in a cross-lingual scenario. The problem is extremely challenging given that we do not have any lecture data in any Indian language.

We attempt to understand the phonotactic variations between English and Indian languages in Section 4.2. Building a bilingual model requires handling bilingual/multilingual text, which is discussed in Section 4.2.1. Further, two speaker adaptation approaches are discussed in Section 4.3 - one using a statistical parametric HMM framework and another using a neural network-based end-to-end framework. The different evaluation techniques and discussions are presented in subsection 4.3.3 and 4.3.4 respectively. Section 4.4 encompasses the summary of the chapter.

4.2 PHONOTACTIC VARIATIONS BETWEEN ENGLISH AND INDIAN LANGUAGES

The task of speaker-adaptation from English to Indian languages without any speaker's data in any Indian language is challenging. This is because English and Indian languages are phonotactically very different. Phonotactics refers to the set of permissible phone sequences in a particular language. Even within Indian languages, the set of rules varies significantly. Indian languages can be categorized into two language families - Indo-Aryan and Dravidian. In Indo-Aryan languages, *schwa* deletion occurs. In contrast, in the case of Dravidian languages, *agglutination* occurs in the scripts (multiple words combined to form a single word, leading to longer words)

(Prakash *et al.*, 2016). English has an entirely different set of phonotactic constraints.

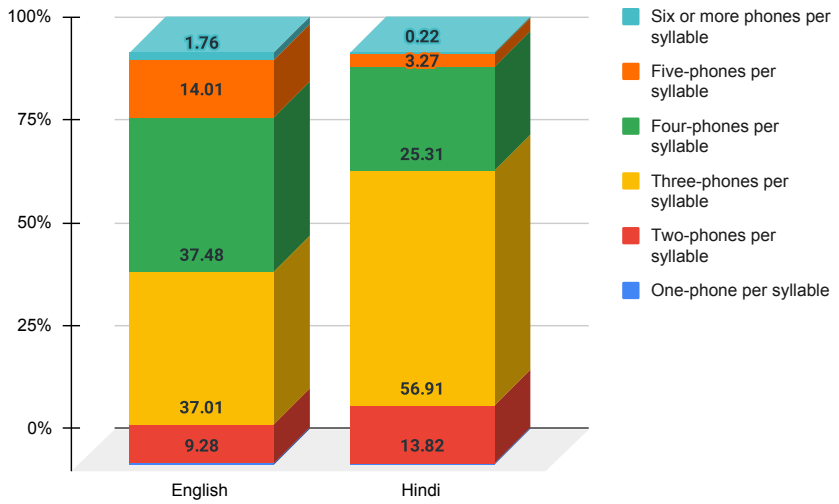


Fig. 4.1: Phonotactic variations between English and Hindi

Statistical analysis of letter clusters by Sen and Samudravijaya (2002) indicates that 'bh' and 'sr' clusters are infrequent in English. In the same way, clusters like 'our' and 'ion' are rare in Indian languages. Indian languages are Akshara-based, and most words/syllables have simple phone clusters (Prakash *et al.*, 2016). English, on the other hand, has complex phone structures. In English, words like *strange* and *sprint* have up to six phones within a single syllable. An analysis of the number of phones within a syllable is shown for English and Hindi in Figure 4.1. For this analysis, approximately 3000 text utterances are considered from English and Hindi datasets from read speech as stated in Section 3.2 and the unique words and syllables are parsed in a common representation. The common representation will be discussed in Section 4.2.1. In Figure 4.1, it is evident that in English, more than 50% of the syllables are made up of four or more phones. However, in the case of Hindi, the majority of the syllables have three or fewer phones. For other Indian languages, the number of phones constituting a syllable remains more or less similar as depicted for Hindi (Prakash *et al.*, 2016). Another phonotactic variation is caused by geminates, where a long or doubled consonant occurs in combination with its shorter counterpart (Davis, 2011). English does not have any geminates (Davis, 2011) whereas geminates are common in Indian languages. Handling these phonotactic differences during voice adaptation is very crucial for robust models.

The phonotactic variations between English and any Indian language are handled in our work by building a bilingual base model. The model is trained by pooling in English and an Indian language data. Since English is seen during training, the phonetic rules and characteristics of English are captured by the base model along with the target language (Hindi or any other Indian language). This helps in finetuning the model parameters using only English data to impart speaker mannerisms in an Indian language.

4.2.1 Handling multilingual text

As discussed previously, to account for the phonotactic and acoustic mismatches during conversion from English to Indian languages, a bilingual (English + Indian language) TTS is trained as the base model. The text must be given in a common representation for training a bilingual model. This is because similar sounds across the two languages need a single representation for training the TTS. For Indian languages, a common

Table 4.1: Illustration of CLS representation of words in different languages

Language	Word	Phone Sequence	Syllable Sequence
English	London	l-a-n-dx-a-n	lan-dxan
Hindi	बिहार	b-i-h-aa-r	बि - हार्
Bengali	পশ্চিমবঙ্গ	p-a-sh-c-i-m-b-a-ng-g	পশ্ - চিম্ - বঙ্
Tamil	தமிழ்நாடு	t-a-m-i-zh-nd-aa-dx-u	த - மிழ் - நா - டு
Gujarati	ગુજરાત	g-u-j-r-aa-t	ગુજ્ - રાત્
Malayalam	കേരള	k-ee-r-a-lx-a	കേ - ര - ള
Kannada	ಕರ್ನಾಟಕ	k-a-r-n-aa-tx-a-k-a	ಕರ್ - ನಾ - ಟ - ಕ

label set (CLS) (Ramani *et al.*, 2013) representation is used to convert the text into the corresponding phone-level representation. This is done using the unified parser (Baby *et al.*, 2016*a*) which generates phoneme and syllable sequences. An English pronunciation dictionary is used to parse the English word. The CLS representation of all the words is first created in a monophone and syllable dictionary. If the words from the training set are not present in the dictionary, they are added to the dictionary

along with their phone-level representations. The addition is done by transliterating the words into an Indian language, manually verifying, and parsing using the unified parser. A sample of words with corresponding phone and syllable level representations is shown in Table 4.1.

4.3 EXPERIMENTS

The overall block diagram of the steps involved in speaker adaptation from read speech using conversational speech data is shown in Figure 4.2. In the diagram, Hindi is shown as an Indian language for training the bilingual model. However, the bilingual model can be trained using multiple Indian languages in combination with English. After the

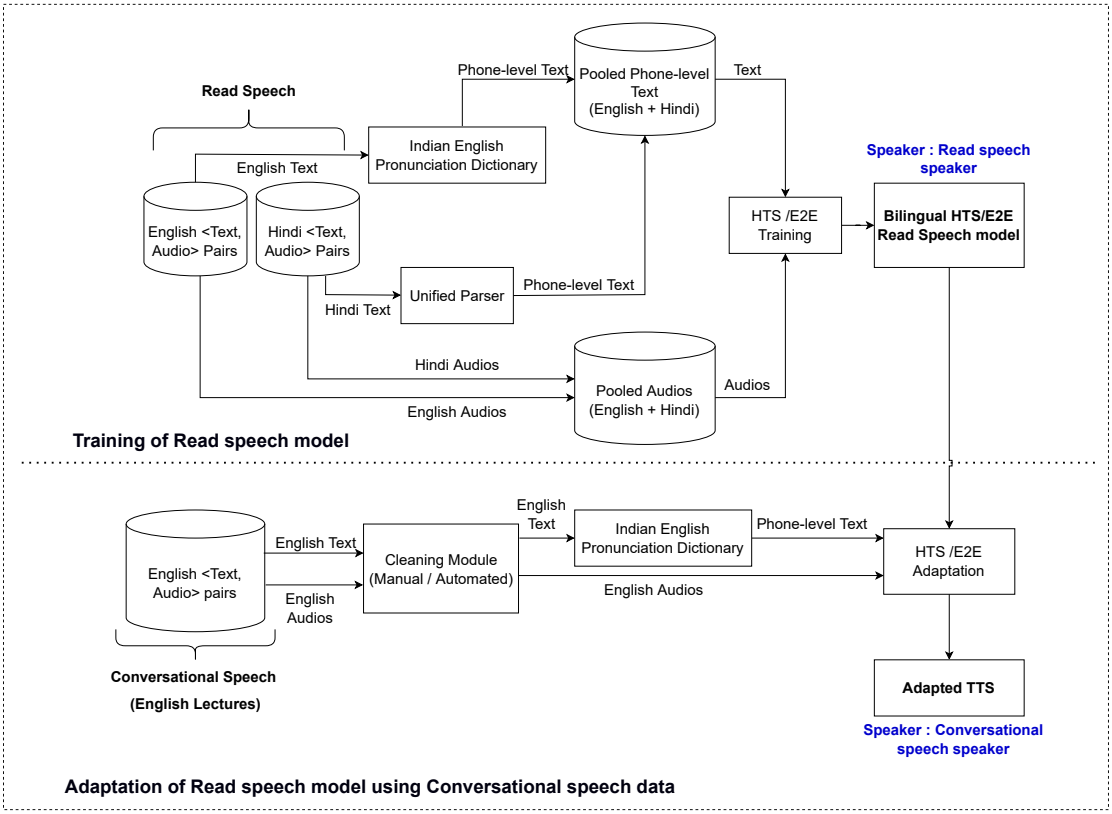


Fig. 4.2: Block diagram of speaker adaptation

English and the Hindi text are parsed, the read-speech audio and the corresponding text are pooled and shuffled. Here shuffling represents reordering of the texts and the audios so that the training data has a good mix of English and Hindi data (text + audio) instead of having data of only a specific language which might add bias in the model during training. Then the text and audio pairs are passed on to the training modules of

the HMM-based speech synthesis system (HTS) or end-to-end (E2E) speech synthesis system. Now, the lecture data, which is in English, is cleaned either manually or using data pruning techniques. The parsed text and the corresponding audios are passed as input to the HTS or E2E adaptation module along with the read-speech model. This adaptation module imparts speaker characteristics to the synthesized audios. The read speech model is trained using English and Hindi data of a native Hindi female speaker represented in Table 4.2. For adaptation, lecture data of CF1 is manually curated to remove disfluencies, ensure proper sentence endings, and uniform syllable and pitch. This manually curated data is used for adaptation using both the frameworks. But, the manual curation stage is time-consuming. Hence, we use the pruning techniques discussed in Section 3.5.1 to obtain about 1 hour of pruned data of CF1 for adaptation. This data is adapted using the E2E framework. The details of experiments performed, along with their naming tags and amount of data used, are given in Table 4.2. The two adaptation techniques – HMM-based speech synthesis (4.3.1) and End-to-End speech synthesis (4.3.2) are discussed below.

Table 4.2: Dataset details with tags for different speaker adaptation experiments

Training			**Adaptation**			**Exp. Tag**
Speaker	**Language**	**Duration (Hrs)**	**Speaker**	**Duration (Hrs)**	**Curation**	
HTS Framework						
RF1	Hindi	8.5	CF1	1/2 (English)	manual	HTS System
	English	8.5				
E2E Framework						
RF1	Hindi	8.5	CF1	1/2 (English)	manual	E2E_manual
	English	8.5				
RF1	Hindi	8.5	CF1	1 (English)	pruning	E2E_auto
	English	8.5				
RF1	Hindi	8.5	RF5	1/2 (English)	read speech	E2E_read
	English	8.5				
RF3	Kannada	8.5	CF1	1 (English)	pruning	E2E_auto_kan
	English	8.5				
RF1	Hindi	8.5	CF2	1 (English)	pruning	E2E_auto_spk2
	English	8.5				
RF3	Kannada	8.5	CF2	1 (English)	pruning	E2E_auto_spk2_kan
	English	8.5				

4.3.1 Speaker adaptation in HTS Framework

The speaker adaptation in HTS takes place along with the training of the base model. There are two steps involved :

- Data Segmentation - The text and the audio are segmented into phones and syllables and the corresponding labels are obtained.
- Training and adaptation - The generated labels are passed to train context-dependent HMM models, build voices and perform speaker adaptive training

Data Segmentation

The syllables and the phones obtained while parsing the text and the audios are passed as input to the segmentation module. Initially, a monophonic model is built using a flat start followed by a forced alignment. To correct the boundaries, HMM models are trained and corrected using group delay segmentation (Baby *et al.*, 2017). The waveforms are spliced at the syllable level, and embedded reestimation is performed iteratively to adapt the monophonic models. Further forced alignment is performed at the syllable level to obtain phone level boundaries (Shanmugam, 2015). The syllable boundary labels and the spectral and excitation parameters from the audios are passed as input to the HTS framework.

System training and adaptation

HMM-based speech synthesis systems (HTS) are one of the most traditional statistical parametric speech synthesis models. They are known to be robust and synthesize good-quality speech even with limited data. HTS provides flexibility in mixed-gender modeling, speaker adaptation, and feature-space adaptive training and is widely used in speech synthesis (Yamagishi *et al.*, 2009).

We use HTS to adapt to the target speaker (CF1 in this case) as a preliminary experiment because it requires a small amount of labeled data for adaptation. This model is referred as the **HTS System** as given in Table 4.2. Figure 4.3 shows a block diagram of the basic steps in HTS training and adaptation. The spectral and the excitation parameters are extracted from the audios and modeled using multistream HMMs (Yamagishi *et al.*, 2009). The first and second derivatives of the static features are taken into consideration. The text is transformed into context-dependent phoneme labels, and context-dependent HMM models are trained. Decision-tree-based context clustering is performed, and a Gaussian probabilistic density function

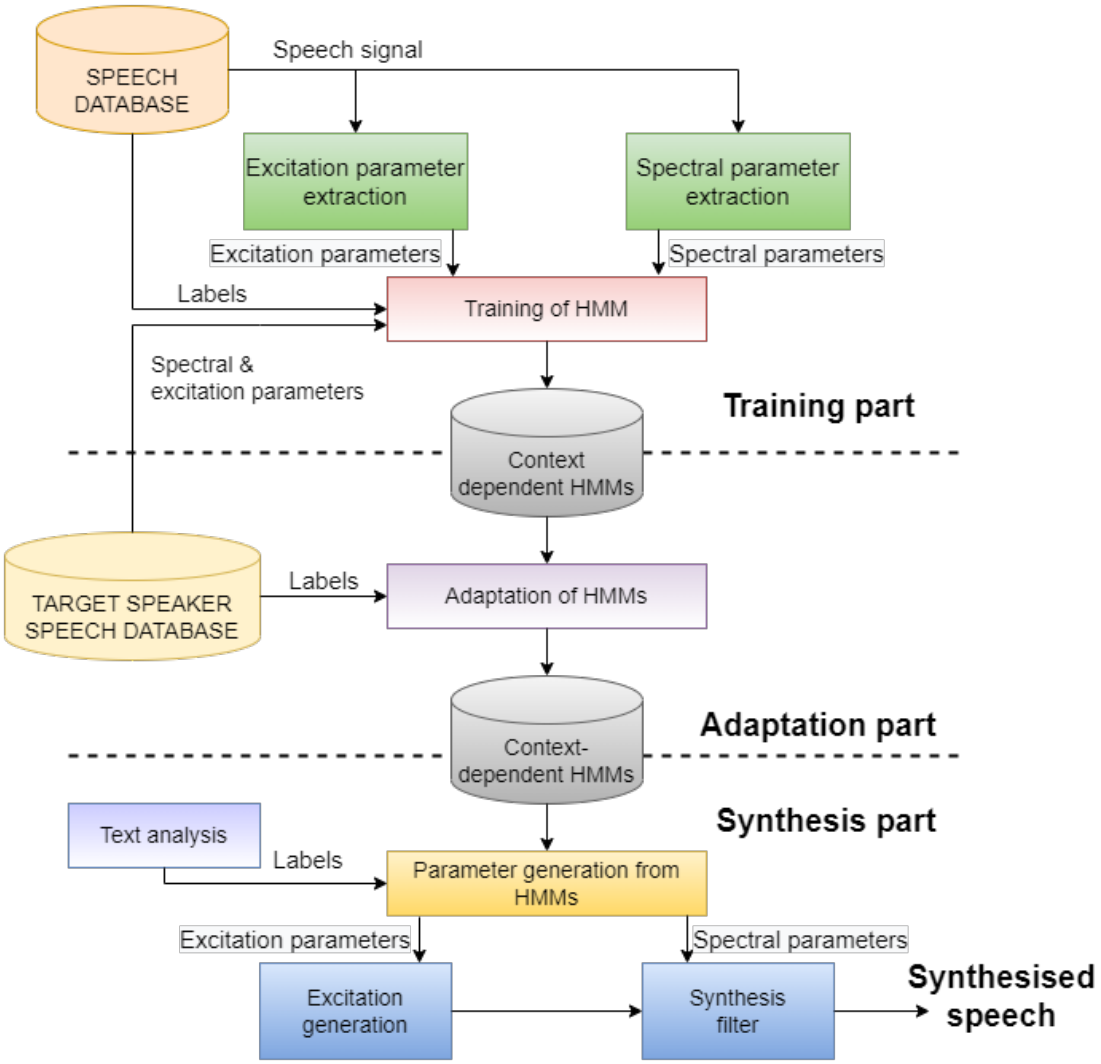


Fig. 4.3: Speaker Adaptation in HTS framework

of averaged voice model is obtained. In our case, we do not have a multi-speaker database for each language; hence we use just a single speaker. The parameters of the average voice model, such as mean vectors of output distributions, are reestimated using the adaptation data in the speaker adaptive training (SAT) stage. The output state distributions for the target speaker are obtained by Maximum likelihood linear regression (MLLR) adaptation. The F0 parameters are also obtained similarly by applying the same algorithm (Junichi, 2006).

During the synthesis stage, the text is converted to phoneme sequences along with their contexts. The spectral and F0 parameters are predicted from the trained HMMs along with the phoneme durations and synthesized using the HTS engine vocoder. The evaluation results of the HTS system are discussed in Section 4.3.3.

4.3.2 Speaker adaptation in E2E framework

The English and the Hindi texts are parsed into phone-level representation using CLS 4.2.1. However, the E2E module maps each character into distinct tokens. The mapping technique proposed by Prakash *et al.* (2019) is used to ensure that the training is purely phonetic. An example of phone-based representation used in E2E is shown in Table 4.3 and the mapping technique is discussed in detail in Appendix C. Training the bilingual read speech model and adaptation using conversational speech data in E2E is similar to the training discussed in Section 3.5.2. The only difference is that x-vectors are used as speaker embeddings in this case to impart speaker characteristics. x-vectors are fixed-length embeddings obtained as the output of a 5-layer Time-Delay Neural Network (TDNN) architecture (Snyder *et al.*, 2018). A block diagram describing the training and adaptation stages in the E2E framework is shown in Figure 4.4. The phone-level text is passed as input to the Tactron2 architecture. The feature extraction module extracts Mel-spectrograms from the audio, and the x-vector module extracts x-vector for each utterance. The x-vector embedding is appended to each encoder state in the Tacotron2 architecture. The network is then trained to learn the mappings between the text and the audio spectrograms using the encoder-decoder architecture with attention. Once the bilingual model is trained, x-vectors are extracted from the conversational data, and the model is finetuned similarly. During testing, the mean x-vector of adaptation data is appended to all the encoder states to synthesize audio in the target speaker's voice. Two variations of E2E adapted systems are trained as a part of this work :

- E2E_manual - The adaptation data is curated manually to maintain uniform syllable rate, pitch, structured sentences, and disfluencies are removed.
- E2E_auto - The adaptation data is curated using pruning techniques discussed in Section 3.5.1.

The details of data used for each experiment in the E2E framework, along with the tags, are given in Table 4.2. More data is used in the case of the automatically pruned model to account for the variations in context and sentence endings which are prevalent in the case of lectures.

Table 4.3: Illustrations of phone based representations used in E2E

Word	Phone-level representation
danger	ड‌Enjar
shorter	शऑrटar
अनुवाद	anuwAd
आधुनिक	Aधunik

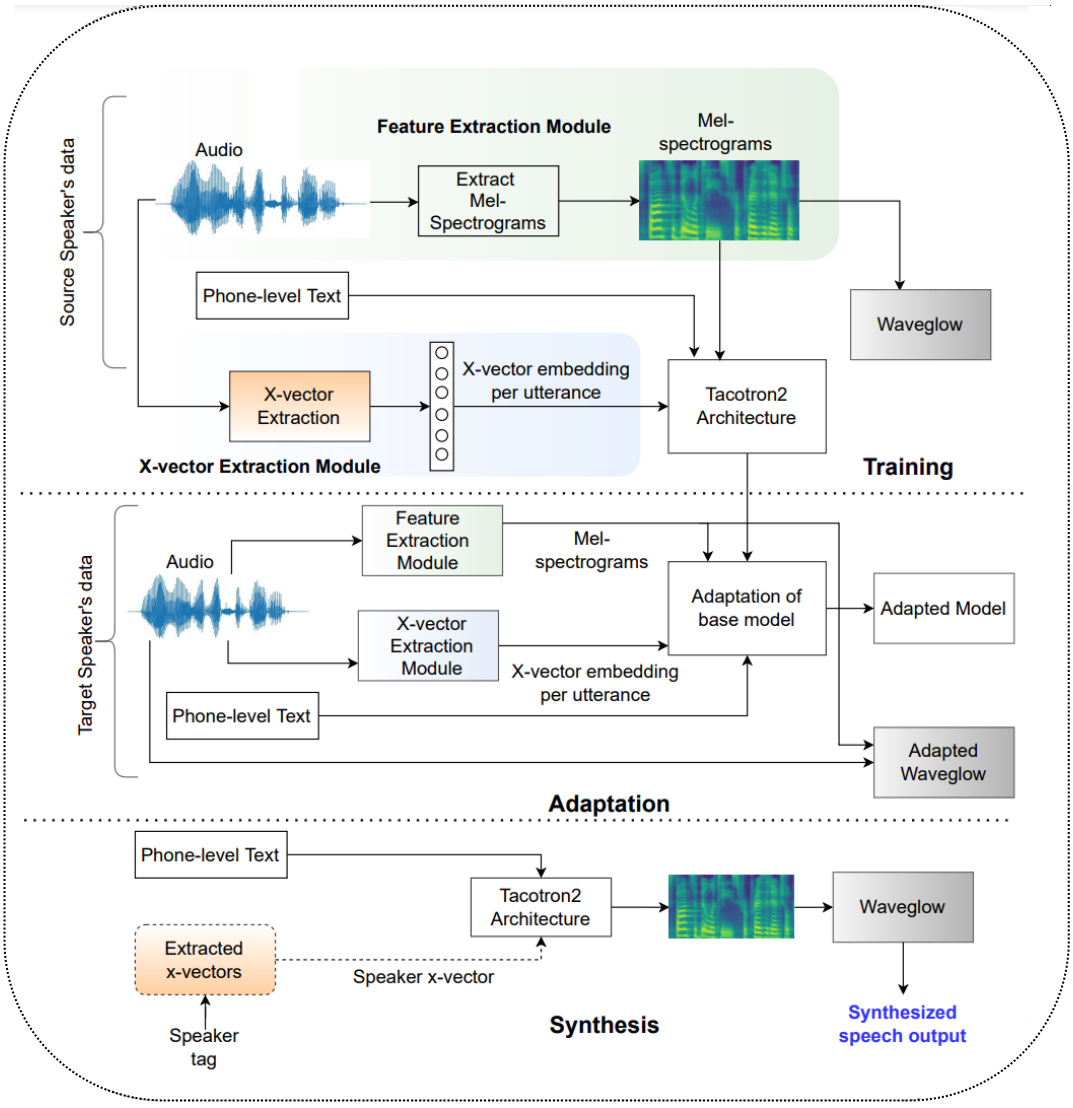


Fig. 4.4: Speaker Adaptation in End-to-End framework

4.3.3 Evaluation

We perform subjective and objective evaluations for monolingual and bilingual (Hindi+English) sentences on the systems - HTS System, E2E_manual, and E2E_auto.

In monolingual synthesis, test sentences are in English. The bilingual Hindi+English test set is a "cross-lingual" scenario, as the original speaker's data is in English, and we are attempting to synthesize bilingual sentences. Bilingual test sentences are obtained by translating the English transcriptions of the lectures into Hindi while retaining the technical terms in English. Two types of subjective evaluations are performed – degraded mean opinion score (DMOS) (Viswanathan and Viswanathan, 2005) and speaker similarity test. For objective evaluations, cosine similarity is computed for monolingual and bilingual sentences. Mel-cepstral Distortion (MCD) (Kubichek, 1993) is computed only for monolingual sentences as MCD requires parallel utterances as we don't have any speaker's utterance in Hindi. The demo audio samples from different speakers and languages can be found in the link (https://www.iitm.ac.in/donlab/preview/test/index.html)

Mel-cepstral Distortion (MCD)

Synthesized utterances generated from the different systems are compared with respect to the original audio using dynamically time-warped (DTW) Mel-cepstral distortion (MCD) scores (Kubichek, 1993). A lower score indicates less distortion, which means more similarity to the original. 20 unseen English sentences from lecture data are considered for the MCD calculation. The MCD scores for the three systems are shown in Figure 4.5.

Cosine Similarity

X-vector embeddings are extracted from the synthesized monolingual and the cross-lingual utterances as well as the original audios. The cosine similarity between the system-generated utterances' x-vectors and the original utterances' x-vectors is computed. Twenty utterances have been considered for calculating the cosine similarity. For monolingual, parallel utterances have been used. In Table 4.4, we observe that the HTS system receives a low similarity score compared to the E2E_manual and E2E_auto systems. A lower similarity score is obtained in a cross-lingual scenario compared to the monolingual case because x-vectors contain some linguistic information (Raj *et al.*, 2019). Figure 4.6 shows T-SNE plots for utterances synthesized using different systems.

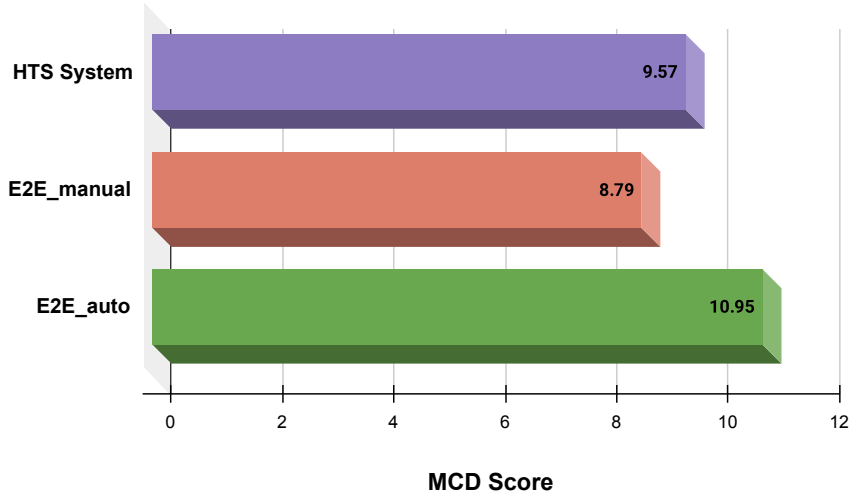


Fig. 4.5: MCD scores for monolingual utterances

The plot correlates closely with the cosine similarity value, where the HTS synthesized embeddings are far apart from the original utterance embeddings. Therefore it has less cosine similarity in both monolingual and cross-lingual cases. E2E_auto and E2E_manual lie somewhere in between the HTS and the original embeddings; therefore are more similar.

Table 4.4: Cosine similarity for different systems

Systems	Monolingual utterances	Cross-lingual utterances
HTS System	0.650	0.594
E2E_manual	0.732	0.698
E2E_auto	0.796	0.754

Speaker Similarity

The speaker similarity test is a subjective evaluation technique extensively used in voice conversion. A set of unseen utterances synthesized using the different systems are presented to the listener along with a few original audios of a particular speaker. Two original audios are presented as references for the target speaker. The listener is asked to rate the audios based on similarity with respect to the reference audios on a scale of 1-5 where 5 indicates most similar. The score is then normalized with respect to the score of the original audios. For speaker similarity, 27 sentences were evaluated (8 from each system + 3 original). The scores for different systems are shown in Figure 4.7

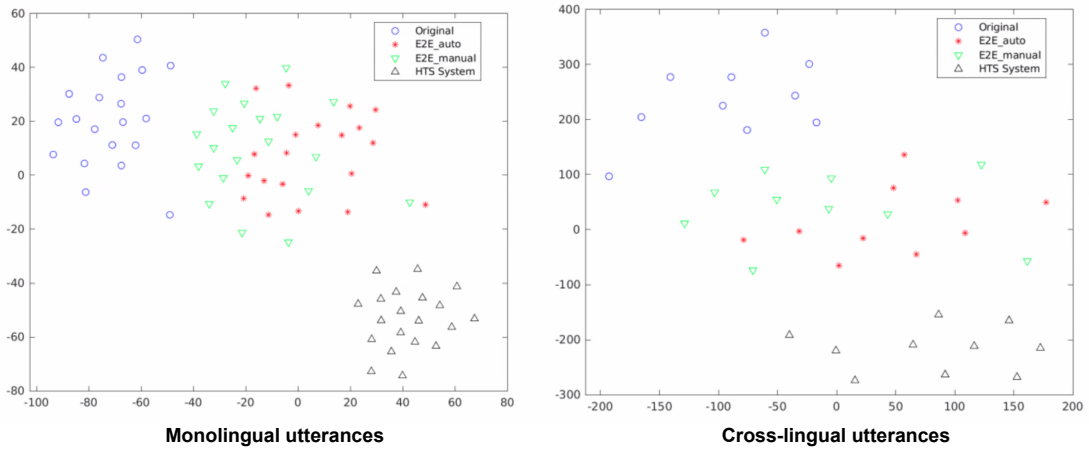


Fig. 4.6: T-SNE plots for monolingual and cross-lingual utterance embeddings synthesized using different systems

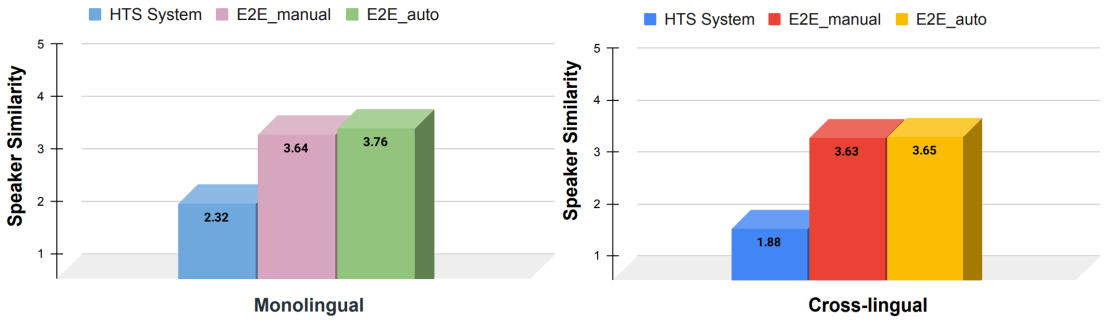


Fig. 4.7: Speaker similarity score for monolingual and cross-lingual utterances

DMOS score

In the DMOS test, listeners rate the quality of the synthesized speech, and the score is normalized with respect to that of the original speech. DMOS test has already been discussed in Section 3.5.2. Seventeen listeners participated in the evaluation and rated 25 sentences (7 from each system + 4 original) each for monolingual and cross-lingual tasks. The DMOS scores for the different systems are shown in Figure 4.8. It is observed that the E2E_manual system receives the highest rating in terms of system intelligibility. The HTS system receives a very poor DMOS score. The possible reasons of low ratings will be discussed in Section 4.3.4.

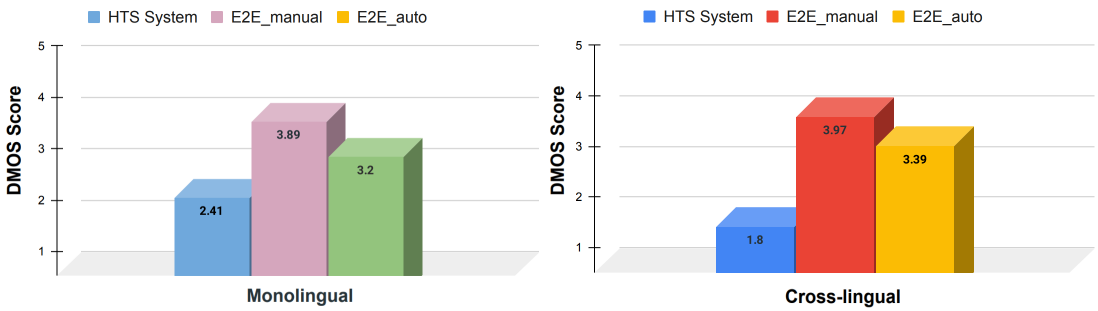


Fig. 4.8: DMOS score for monolingual and cross-lingual utterances

Extension to other Indian languages and other speakers

We attempt to convert the voice of the speaker CF1 to a Dravidian language, Kannada, to verify the scalability of the technique across Indian languages. We have used automatically pruned adaptation data for this task, and the system is named as **E2E_auto_kan** as stated in Table 4.2. A bilingual (English + Kannada) model of speaker RF3 is built in the E2E framework using attention-based Tacotron2 architecture in the same way as discussed in Section 4.3.2. It is then adapted using the pruned adaptation data of speaker CF1. We perform DMOS (3.5.2) and speaker similarity (4.3.3) tests, and the result is presented in Table 4.5. The DMOS test is performed using 8 unseen Kannada sentences synthesized using the adapted model and two original sentences and is evaluated by 25 native Kannada speakers. For the speaker similarity test, native or non-native speakers participated since speaker similarity is language agnostic. 8 unseen Kannada sentences and two original English sentences of CF1 are used for the evaluation and are rated by 15 evaluators. We attempt to verify that

Table 4.5: Cross-lingual (Kannada) DMOS and speaker similarty scores

System	DMOS Score	Speaker Similarity score
E2E_auto_kan	**3.03**	**3.359**

pruning and cross-lingual adaptation is extendable to other speakers. For this, another speaker's data (CF2) has been considered for adaptation, and the same experiments are performed using bilingual Hindi and the bilingual Kannada models stated previously. The data used for this experiment is given in Table 4.2 with the tag **E2E_auto_spk2** and **E2E_auto_spk2_kan**. 20 unseen sentences are synthesized in English and Hindi using the E2E_auto_spk2, and 20 unseen sentences are synthesized in Kannada using the

system E2E_auto_spk2_kan. Since subjective evaluations are cumbersome, objective evaluations such as MCD and cosine similarity have been performed. MCD has been performed only using the unseen sentences in English, and it is found to be **11.99** which is close to E2E_auto 4.3 of speaker CF1, which was **10.98**. We also perform cosine similarity and compare the results of CF2 with CF1, and the comparative results are presented in Table 4.6.

Table 4.6: Cosine similarity values for monolingual and cross-lingual utterances of different speakers

Speakers	**English**	**Hindi**	**Kannada**
CF1	0.796	0.754	0.666
CF2	0.795	0.767	0.746

4.3.4 Discussion

Why speaker adaptation fails in HTS framework?

Although the MCD score for the HTS system is low, indicating less distortion, the cosine similarity and the subjective evaluations speak differently. As evident from Section 4.3.3, the HTS adapted system receives a poor score in speaker similarity and DMOS even after using manually curated data for adaptation. The synthesized audios obtained using this system seem to be complete sentences without repetitions but of poor quality. The critical reasons for the poor quality of synthesized samples could be one or more of the following:

- The pitch is averaged during training and adaptation in HTS, and our data being conversational with varying pitch might not be able to form a good distribution for random sample generation.
- As stated by Yamagishi *et al.* (2009), a multi-speaker database is expected for robust speaker adaptation since it provides more generalization to the model. But we do not have a multi-speaker database for Indian languages. Hence the synthesis quality degrades.
- Due to multiple tying of states in the case of HMMs, the audios seem to be muffled. This is because we are training bilingual models, and the phonotactics of English and Indian languages differ remarkably.

All these reasons affect the quality and speaker similarity in the HTS framework.

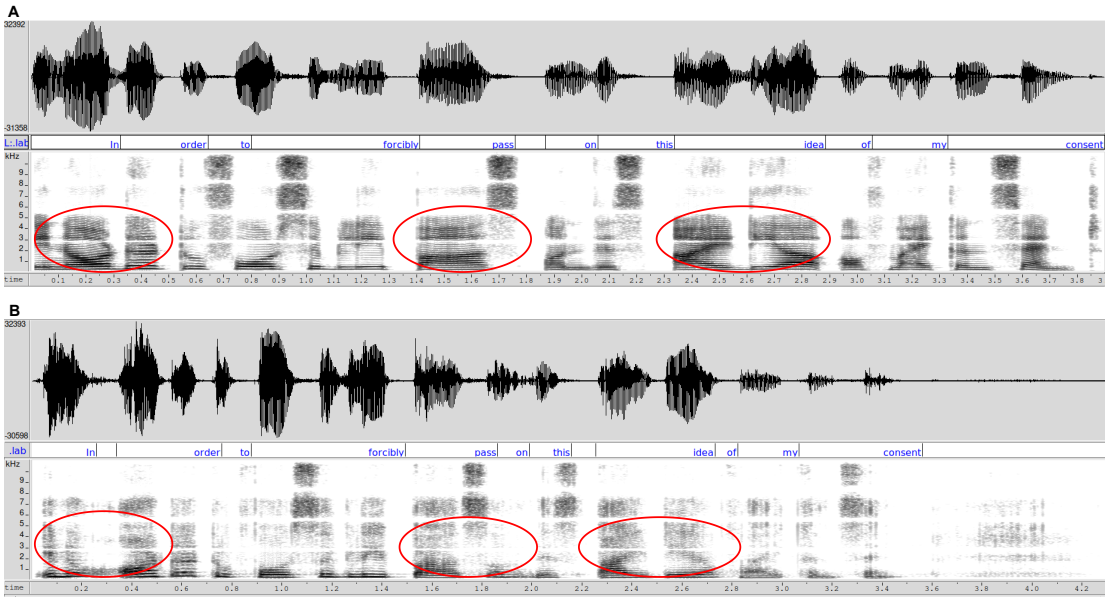


Fig. 4.9: Spectrogram analysis of an utterance with adaptation in read speech (A) and conversational speech (B)

Speaker Adaptation in read and conversational speech : A comparative analysis

Prakash and Murthy (2020) states that speaker adaptation using read speech in Indian languages is feasible with a DMOS score of 4.41 and speaker similarity of 3.95 using 7 mins of adaptation data. But using conversational speech data for adaptation, the maximum achievable DMOS was found to be 3.65 and a speaker similarity of 3.39. On the informal evaluation of the synthesized sentences, we notice that the audios generated using conversational speech adaptation result in trembling and muffling of the voice. We try to introspect the reason for this by synthesizing 20 common sentences across a read speech adapted model E2E_read and a conversational speech adapted model E2E_auto_spk2 as given in Table 4.2. A sample utterance is synthesized using both the models, and analysis is discussed.

In Figure 4.9, we observe that in read speech adaptation, the spectrogram captures more information than adaptation in conversational speech, as shown by the markers. The formants are more clearly visible in the first case. Even in the time domain representation of the audio, the words ‘In’, ‘pass’, and ‘idea’, which contain vowels, have distinct representations for read speech. In the case of conversational speech, the vowel waveforms are not as clear as in the first case.

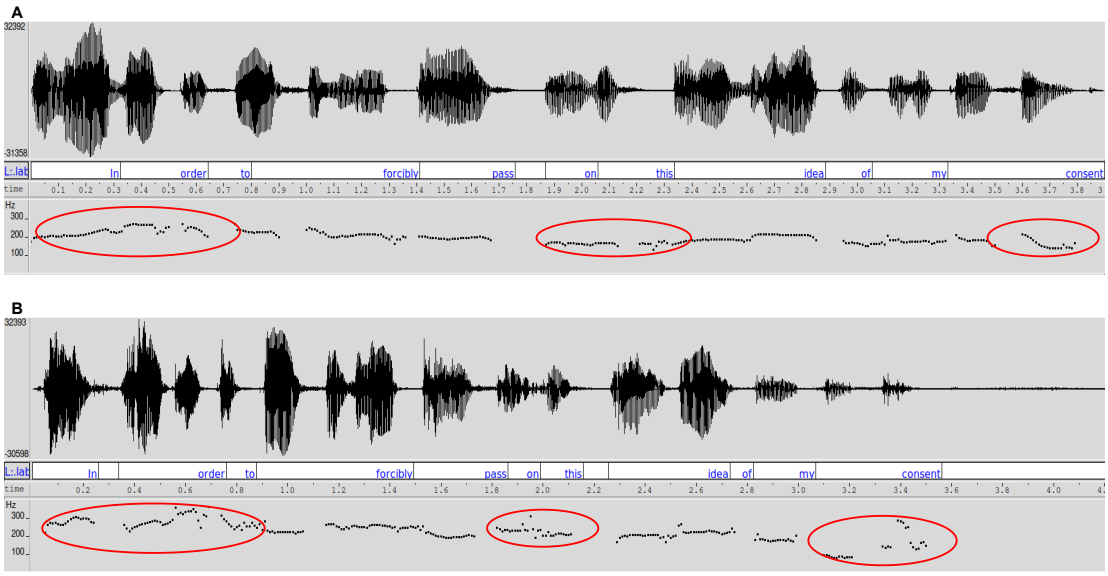


Fig. 4.10: Pitch contour analysis of an utterance with adaptation in read speech (A) and conversational speech (B)

In Figure 4.10, we find that the pitch contours are highly discontinuous in the case of conversational speech adaptation, especially in the region of phrases ‘in order to’ , ‘on this ’, and ‘consent’. In read speech adaptation, the pitch contours are smooth and uniform. As discussed in Section 3.3.2, pitch fluctuations were visible at the end of an utterance. A similar trend is seen even after adapting a base read speech model using conversational speech data. Due to the lack of context and abrupt sentence endings in conversational speech, it gets very difficult to detect a sentence’s end, resulting in arbitrary pitch at the end of the sentence. There is a smooth transition of the pitch in the case of read speech, indicating the end of the sentence. These pitch fluctuations might cause trembling in the synthesized utterances in conversational speech adaptation. In the short-term energy plots of Figure 4.11, the silence boundaries of the two utterances are marked in red. We notice that the silence regions are carefully represented with low energy in the case of read speech adapted utterance. In contrast, the silence regions have some high energy components in utterance adapted using conversational speech. Also, there is an extended silence region at the end of the utterance synthesized using lecture data. But some energy can be seen even in those regions that might influence the utterance’s intelligibility.

In Figure 4.12 we observe that the short-term energy fluctuations are higher in the

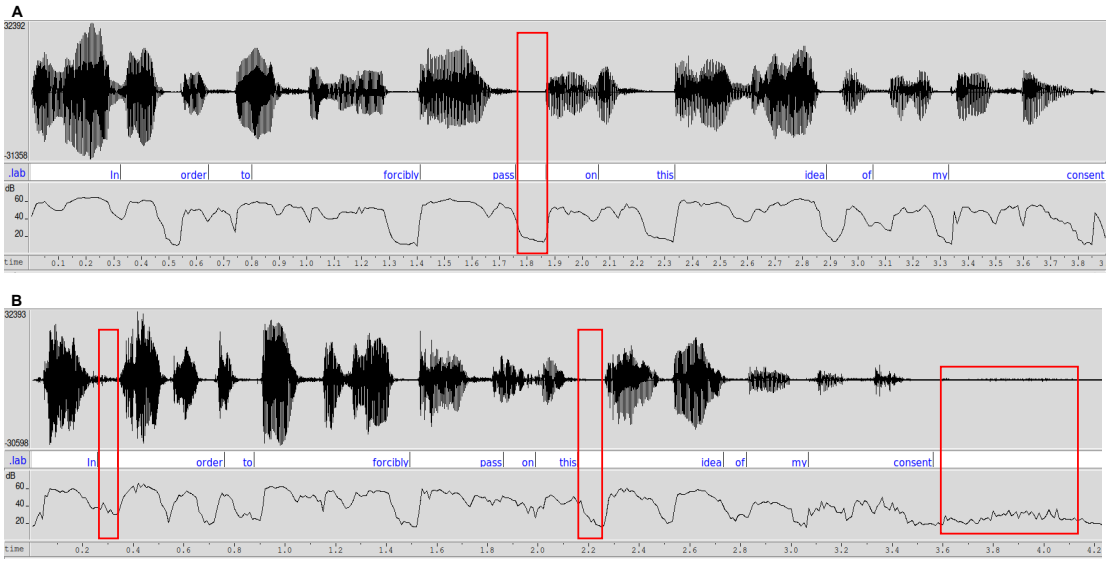


Fig. 4.11: Silence analysis of an utterance with adaptation in read speech (A) and conversational speech (B)

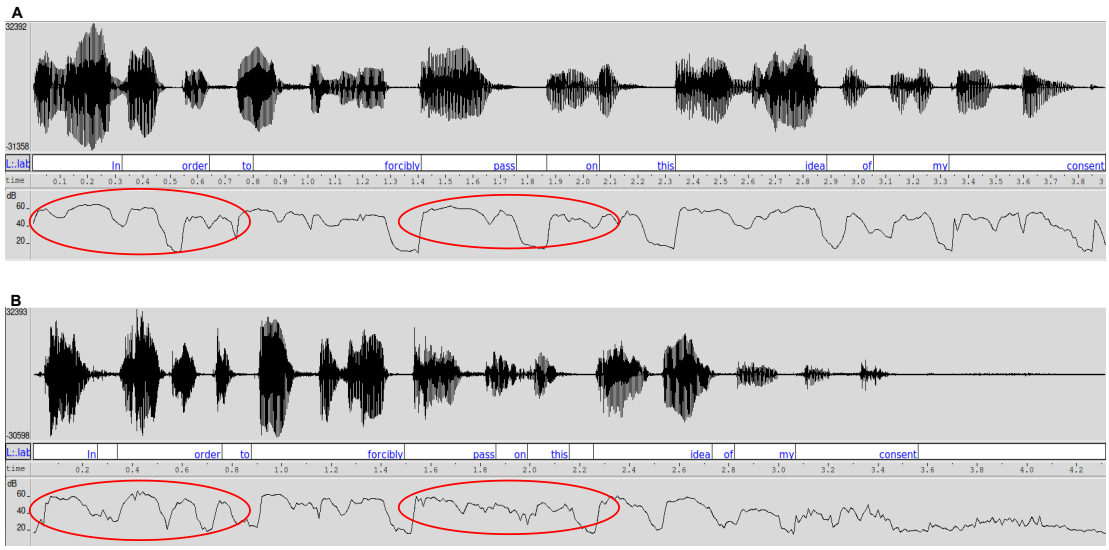


Fig. 4.12: Short term energy analysis of an utterance with adaptation in read speech (A) and conversational speech (B)

case of conversational speech adapted utterance. In comparison the utterance adapted using read speech shows that STE transitions are smooth.

All these reasons might account for the trembling voice when we use conversational speech data for voice adaptation.

4.4 SUMMARY

In this chapter, we discussed techniques for handling bilingual or multi-lingual text. We used conversational lecture data in English for speaker adaptation in a low-resource scenario to generate voices in Indian languages. The base model is built using read speech and adapted using English data. We show how the model can scale to multiple speakers and languages. However, the intelligibility is still not at par with that of read speech adapted models. We further introspect the possible causes of this degradation in synthesis quality. The experiments and the results form a foundational block for further exploration in conversational speech voice adaptation in Indian languages with less amount of data.

CHAPTER 5

CONCLUSION

5.1 SUMMARY

The thesis is an attempt to understand the characteristics of conversational speech primarily in the context of building real-time conversational TTS systems. Further, a more ambitious task of voice adaptation using lecture data has been attempted. The demand for spontaneous and expressive TTS has increased tremendously with the rapid increase in chatbots, voice assistants, and other online platforms which provide real-time user interfaces. Imparting human-like characteristics into the system-generated voice is a new challenge. It is imperative to understand the fundamental difference between read speech and conversational speech to build robust TTS models. Read speech systems have already achieved a human-like quality. However, it generates impassive speech, which is not user interactive. The vision is to achieve a human-like quality with prosodic modulations and expressions in system-generated voice. Our work primarily focuses on classroom lecture data and attempts to use this conversational data to generate the speaker's voice in a target Indian language. Several interesting analyses in due course conclude that conversational speech voice adaptation is much harder compared to read speech. Attempting conversational speech voice adaptation in a cross-lingual scenario to generate Indian languages, which are low-resourced and phonotactically different, is a tall order.

The predominant factors which differentiate read and conversational speech have been identified. We observe how syllable rate, pitch, SNR, and ASR errors percolate into system training and affect the model's robustness. A pruning module has been introduced in Chapter 3 based on a few parameters to prune the inadequate data before training/adaptation. Although the model is still not comparable to read-speech systems even after pruning, the synthesized audios, in some instances, sound more natural with voice modulations. This is encouraging as it demonstrates that building text-to-speech systems using classroom lectures is feasible. However, more research in this direction

is needed to identify if any other factors degrade the model intelligibility apart from the ones discussed in Chapter 3.

The second part of the thesis discusses voice adaptation approaches using a minimum amount of conversational speech data in Chapter 4. A robust read-speech bilingual model is trained as the base model to handle phonotactic variations. A study on phonotactics between English and Hindi shows the variability in these two languages. The HTS-adapted system generates complete sentences without repetitions as the modeling is done at the phone level. However, the intelligibility and speaker similarity are inconsistent with the end-to-end adapted models. The reasons for the deterioration in quality are further analyzed. A comparison between speaker adaptation in read speech and conversational speech is performed. It suggests that the intricate details in terms of pitch contour discontinuity, energy fluctuations, etc., which are seen in the original utterances of conversational speech, are reflected even in the synthesized sentences. The evaluations demonstrate that a DMOS of 3.39 and speaker similarity of 3.65 is best achieved so far using automated techniques in cross-lingual adaptation.

The thesis forms the foundation for any research encompassing conversational text-to-speech synthesis, voice conversion, and voice adaptation. Results of cross-lingual voice adaptation to Indian languages using only the speaker's English data motivate further exploration in this field. The thesis also shows what has been achieved so far in voice adaptation in the speech-to-speech pipeline for lecture transcreation in Indian languages.

5.2 CRITICISM OF THE THESIS

Transformer-based architecture like Fastspeech and Fastspeech2 can be tried out. Since Fastspeech2 explicitly models the pitch, it can be an added advantage in modeling the pitch fluctuations in conversational speech. Global style token (GST) in addition to the x-vector embeddings can be tried for adding expressions and speaker mannerisms in the synthesized speech.

In the HTS framework, a multi-speaker database for each Indian language can be used for training the base model to create robust average voice models leading to better speaker adaptation. Currently, we do not have a multi-speaker dataset for each Indian

language.

5.3 SCOPE OF FUTURE RESEARCH

The work discussed in this thesis encompasses conversational speech, cross-lingual voice conversion and adaptation, and Indian language speech synthesis. Hence it can be branched further into various interesting research directions.

More exploration in conversational speech is needed to build robust speech synthesis models that mimic human expressions and prosody. Since the availability of accurate transcriptions is a significant bottleneck in this problem, transcription-free or speech-to-speech voice conversion techniques can be attempted. Acoustic unit discovery for conversational speech can help remove transcription errors and generate a direct mapping between the source and target acoustics. Spectral mapping using linear-predictive spectrum and linear predictive residual can also be tried out for generating the target voice. ASR, in tandem with the TTS module, can be tried out for cross-lingual voice adaptation.

Disfluencies and filler words often add to the challenges in spontaneous speech. Automatic disfluency identification and removal techniques can be explored to curate the data further. Analysis of the speaking rate of the professors during a lecture can be analyzed. This can help in position-driven adaptation for more naturalness in a particular region of a lecture after video transcreation. This work forms the foundation to demonstrate that personalized TTS systems can be built using pruning techniques. The work is extendable to multiple Indian languages and multiple speakers.

APPENDIX A

Conversational Speech Dataset

Speaker_ID	Speaker	Course No.	Course Name	Duration (Hrs)
CM1	Madhavan Mukund	106106145	Programming, Data Structures & Algorithms using Python	1/2
CM2	Partha Pratim Das	106105151	Programming in C++	1/2
CM3	Joydeep Dutta	111104085	Basic Calculus for Engineers, Scientists and Economists	1/2
CM4	M.R Shenoy	115102103	Semiconductor Optoelectronics	1/2
CM5	Arjun Ghosh	109102156	Text, Textuality and Digital Media	1/2
CM6	Mitesh M. Khapra	106106184	Deep Learning Part-1	1/2
CM7	S.R. Kale	112102255	Thermodynamics	1/2
CM8	K. Ramesh	112106065	Engineering Fracture Mechanics	1/2
CM9	Arun K.Tangirala	103106120	Introduction to Statistical Hypothesis Testing	1/2
CM10	Ashish Saxena	109104136	Development of Sociology in India	1/2
CM11	Girish Kumar	108101092	Antennas	≈ 43

Speaker_ID	Speaker	Course No.	Course Name	Duration (Hrs)
CF1	Deepa Venkatish	108106167	Fiber Optic Communication Technology	≈ 45
CF2	Rashmi Gaur	109107154	Body Language : Key to Professional Success	≈ 22
CF3	Kamlesh Singh	109102157	Positive Psychology	1/2
CF4	Sneha Singh	112107290	Acoustic Materials and Metamaterials	1/2
CF5	Jhumkee Iyengar	109104109	Understanding Design Thinking & People Centred Design	1/2
CF6	Meenakshi D'Souza	106101163	Software Testing	≈ 40
CF7	Vatsala Misra	121104005	Introduction to Japanese Language And Culture	1/2
CF8	Sujatha Srinivasan	112106248	Mechanics of Human Movement	1/2
CF9	Rinku Mukherjee	112106190	Introduction to Boundary Layers	1/2
CF10	Merin Simi Raj	109106171	Literary Criticism (From Plato to Leavis)	1/2

APPENDIX B

Common Label Set

Sl. No.	Label	Hindi	Gujarati	Tamil	Kannada
1	a	अ	અ	அ	ಅ
2	ax	-	ઑ	-	-
3	aa	आ	આ	ஆ	ಆ
4	axx	-	-	-	-
5	i	इ	ઇ	இ	ಇ
6	ii	ई	ઈ	ஈ	ಈ
7	u	उ	ઉ	உ	ಉ
8	eu	-	-	உ	◌್
9	uu	ऊ	ઊ	ஊ	ಊ
10	rq	ऋ,ॠ	-	-	ಋ,ೠ
11	e	-	-	எ	ಎ
12	ee	ए	એ	ஏ	ಏ
13	ea	-	-	-	ಎ
14	ei	ऐ	ઐ	-	-
15	ai	-	-	ஐ	ಐ
16	oi	-	-	-	-
17	o	ओ	ઓ	ஒ	ಒ
18	oo	-	-	ஓ	ಓ
19	ae	-	ઍ	-	-
20	au	-	-	ஔ	ಔ
21	ou	औ	ઔ	-	-
22	k	क	ક	க	ಕ
23	kh	ख	ખ	-	ಖ

Sl. No.	Label	Hindi	Gujarati	Tamil	Kannada
24	g	ग	ગ	கv	ಗ
25	gh	घ	ઘ	-	ಘ
26	ng	ङ	ઙ	ங	ಙ
27	c	च	ચ	ச	ಚ
28	ch	छ	છ	-	ಛ
29	cx	-	-	-	-
30	j	ज	જ	ஜ	ಜ
31	jh	झ	ઝ	-	ಝ
32	jx	-	-	-	-
33	nj	ञ	ઞ	ஞ	ಞ
34	tx	ट	ટ	ட	ಟ
35	txh	ठ	ઠ	-	ಠ
36	dx	ड	ડ	டv	ಡ
37	dxh	ढ	ઢ	-	ಢ
38	nx	ण	ણ	ண	ಣ
39	t	त	ત	த	ತ
40	th	थ	થ	-	ಥ
41	d	द	દ	தv	ದ
42	dh	ध	ધ	-	ಧ
43	n	न,ऩ	ન	ந,ன	ನ
44	nd	-	-	ந	-
45	p	प	પ	ப	ಪ
46	ph	फ	ફ	-	ಫ

Sl. No.	Label	Hindi	Gujarati	Tamil	Kannada
47	b	ब	બ	பv	ಬ
48	bh	भ	ભ	-	ಭ
49	m	म	મ	ம	ಮ
50	y	य,य़	ય	ய	ಯ
51	r	र,ऱ	ર	ர	ರ
52	l	ल	લ	ல	ಲ
53	lx	-	ળ	ள	ಳ
54	w	व	વ	வ	ವ
55	sh	श	શ	-	ಶ
56	sx	ष	ષ	ஷ	ಷ
57	s	स	સ	ஸ	ಸ
58	h	ह	હ	ஹ	ಹ
59	kq	क़	-	-	-
60	khq	क़	-	-	-
61	gq	ग़	-	-	-
62	z	ज़	-	-	-
63	jhq	झ़	-	-	-
64	dxq	ड़	-	-	-
65	dxhq	ढ़	-		-
66	dhq	-	-	-	-
67	f	फ़	-	்ப	-
68	bq	-	-	-	-
69	yq	-	-	-	-

Sl. No.	Label	Hindi	Gujarati	Tamil	Kannada
70	nq	-	-	-	-
71	rx	-	-	ற	ಱ
72	sq	-	-	-	-
73	zh	-	-	ழ	-
74	nxh	-	-	-	-
75	nh	-	-	-	-
76	mh	-	-	-	-
77	rh	-	-	-	-
78	lh	-	-	-	-
79	wh	-	-	-	-
80	q	◌ॅ	◌ૅ	-	◌ೀ
81	hq	◌ः	◌ઃ	-	◌ಃ
82	mq	◌ॅं	◌ૅં	-	-
83	x	-	-	-	-

APPENDIX C

Phone mapping technique for End-to-End TTS Systems

CLS Notation	Single - character Notation
aa	A
axx	अ
ii	I
uu	U
ee	E
oo	O
nn	N
ae	ऍ
ag	ऽ
au	औ
ax	ऑ
bh	B
ch	C
dh	ध
dx	ड
dxh	ढ
dxhq	ढ़
dxq	ड़
ei	ऐ
ai	ऐ
eu	ॅ
gh	घ
gq	ग़
hq	H
jh	J
kh	ख

CLS Notation	Single - character Notation
khq	ख़
kq	क़
ln	ൾ
lw	ൿ
lx	ഋ
mq	M
nd	न
ng	ङ
nj	ञ
nk	़
nw	ൺ
nx	ण
ou	औ
ph	P
rq	R
rqw	ॠ
rw	ർ
rx	ऱ
sh	श
sx	ष
th	थ
tx	ट
txh	ठ
wv	W
zh	Z

REFERENCES

1. **Abe, M.**, **S. Nakamura**, **K. Shikano**, and **H. Kuwabara** (1988). Voice conversion through vector quantization. *ICASSP 1988 - 1988 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, **1**, 655–658, doi:10.1109/ICASSP.1988.196671.

2. **AiutoConsulting** (2020). Challenges faced by vernacular medium students. *https://aiutoconsulting.in/challenges-faced-by-vernacular-medium-students/*.

3. **Andersson, S.**, **J. Yamagishi**, and **R. A. Clark** (2012). Synthesis and evaluation of conversational characteristics in HMM-based speech synthesis. *Speech Communication*, **54**(2), 175–188. ISSN 0167-6393, doi:10.1016/j.specom.2011.08.001.

4. **Baby, A.**, **N. Nishanthi**, **A. L. Thomas**, and **H. A. Murthy** (2016*a*). A Unified Parser for Developing Indian Language Text to Speech Synthesizers. *International Conference on Text, Speech, and Dialogue*, 514–521, doi:10.1007/978-3-319-45510-5_59.

5. **Baby, A.**, **J. J. Prakash**, **R. Vignesh**, and **H. A. Murthy** (2017). Deep Learning Techniques in Tandem with Signal Processing Cues for Phonetic Segmentation for Text to Speech Synthesis in Indian Languages. *Proc. Interspeech 2017*, 3817–3821, doi:10.21437/Interspeech.2017-666.

6. **Baby, A.**, **A. L. Thomas**, **N. L. Nishanthi**, and **H. A. Murthy** (2016*b*). Resources for Indian languages. *Community-based Building of Language Resources (International Conference on Text, Speech and Dialogue)*, 37–43. URL https://www.tsdconference.org/tsd2016/download/cbblr16-850.pdf.

7. **Baevski, A.**, **Y. Zhou**, **A. Mohamed**, and **M. Auli** (2020). wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. *Advances in Neural Information Processing Systems*, **33**, 12449–12460. URL https://proceedings.neurips.cc/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf.

8. **Bartlett, S.**, **G. Kondrak**, and **C. Cherry** (2009). On the syllabification of phonemes. *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 308–316, doi:10.3115/1620754.1620799.

9. **Batliner, A.**, **R. Kompe**, **A. Kießling**, **E. Nöth**, and **H. Niemann** (1995). Can You Tell Apart Spontaneous and Read Speech if You Just Look at Prosody? *Speech Recognition and Coding*, 321–324, doi:10.1007/978-3-642-57745-1_47.

10. **Baumann, T.**, **C. Kennington**, **J. Hough**, and **D. Schlangen** (2017). Recognising Conversational Speech: What an Incremental ASR Should Do for a Dialogue System

and How to Get There. *Dialogues with Social Robots: Enablements, Analyses, and Evaluation*, 421–432, doi:10.1007/978-981-10-2585-3_35.

11. **Chen, M.**, **X. Tan**, **B. Li**, **Y. Liu**, **T. Qin**, **S. Zhao**, and **T.-Y. Liu** (2021). AdaSpeech: Adaptive Text to Speech for Custom Voice. *International Conference on Learning Representations (ICLR) 2021*, 1–10. URL https://openreview.net/forum?id=Drynvt7gg4L.

12. **Chou, J.** and **H. Lee** (2019). One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization. *Proc. Interspeech 2019*, 664–668, doi:10.21437/Interspeech.2019-2663.

13. **Chou, J.**, **C. Yeh**, **H. Lee**, and **L. Lee** (2018). Multi-target Voice Conversion without Parallel Data by Adversarially Learning Disentangled Audio Representations. *Proc. Interspeech 2018*, 501–505, doi:10.21437/Interspeech.2018-1830.

14. **Davis, S.** (2011). Geminates. *The Blackwell Companion to Phonology*, 1–25, doi:10.1002/9781444335262.wbctp0037.

15. **Deo, R. S.** and **P. S. Deshpande** (2014). Pitch contour modelling and modification for expressive marathi speech synthesis. *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2455–2458, doi:10.1109/ICACCI.2014.6968430.

16. **Desai, S.**, **A. W. Black**, **B. Yegnanarayana**, and **K. Prahallad** (2010). Spectral Mapping Using Artificial Neural Networks for Voice Conversion. *IEEE Transactions on Audio, Speech, and Language Processing*, **18**(5), 954–964, doi:10.1109/TASL.2010.2047683.

17. **Dufour, R.**, **V. Jousse**, **Y. Estève**, **F. Béchet**, and **G. Linarès** (2009). Spontaneous Speech Characterization and Detection in Large Audio Database. *13-th International Conference on Speech and Computer (SPECOM 2009)*, 41–46. URL https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.627.4704&rep=rep1&type=pdf.

18. **Erro, D.** and **A. Moreno** (2007). Frame Alignment Method for Cross-Lingual Voice Conversion. *Proc. Interspeech 2007*, 1969–1972, doi:10.21437/Interspeech.2007-551.

19. **Hunt, A. J.** and **A. W. Black** (1996). Unit selection in a concatenative speech synthesis system using a large speech database. *ICASSP 1996 - 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings (ICASSP)*, **1**, 373–376, doi:10.1109/ICASSP.1996.541110.

20. **Ito, K.** and **L. Johnson** (2017). The LJ Speech Dataset. https://keithito.com/LJ-Speech-Dataset/.

21. **Junichi, Y.** (2006). Average-Voice-Based Speech Synthesis. *Ph. D. Thesis, Tokyo Institute of Technology*. URL https://www.cs.cmu.edu/~srallaba/pdfs/jy_phd.pdf.

22. **Kain, A.** and **M. W. Macon** (1998). Spectral voice conversion for text-to-speech synthesis. *ICASSP 1998 - 1998 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, **1**, 285–288, doi:10.1109/ICASSP.1998.674423.

23. **Kameoka, H.**, **T. Kaneko**, **K. Tanaka**, and **N. Hojo** (2018). StarGAN-VC: Non-parallel Many-to-many Voice Conversion Using Star Generative Adversarial Networks. *2018 IEEE Spoken Language Technology Workshop (SLT)*, 266–273, doi:10.1109/SLT.2018.8639535.

24. **Kaneko, T.** and **H. Kameoka** (2018). CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks. *2018 26th European Signal Processing Conference (EUSIPCO)*, 2100–2104, doi:10.23919/EUSIPCO.2018.8553236.

25. **Kawahara, H.** (2006). STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds. *Acoustical Science and Technology*, **27**(6), 349–353, doi:10.1250/ast.27.349.

26. **Kim, C.** and **R. M. Stern** (2008). Robust Signal-to-Noise Ratio Estimation based on Waveform Amplitude Distribution Analysis. *Proc. Interspeech 2018*, 2598–2601, doi:10.21437/Interspeech.2008-644.

27. **Kim, J.-H.**, **S.-H. Lee**, **J.-H. Lee**, **H.-G. Jung**, and **S.-W. Lee** (2021). GC-TTS: Few-shot Speaker Adaptation with Geometric Constraints. *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1172–1177, doi:10.1109/SMC52423.2021.9658830.

28. **Kubichek, R.** (1993). Mel-cepstral distance measure for objective speech quality assessment. *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, **1**, 125–128, doi:10.1109/PACRIM.1993.407206.

29. **Meteer, M.** and **R. Iyer** (1996). Modeling conversational speech for speech recognition. *Conference on Empirical Methods in Natural Language Processing*. URL https://aclanthology.org/W96-0204.pdf.

30. **Mohammadi, S. H.** and **T. Kim** (2018). Investigation of Using Disentangled and Interpretable Representations for One-shot Cross-lingual Voice Conversion. *Proc. Interspeech 2018*, 2833–2837, doi:10.21437/Interspeech.2018-2525.

31. **Moss, H. B.**, **V. Aggarwal**, **N. Prateek**, **J. González**, and **R. Barra-Chicote** (2020). BOFFIN TTS: Few-Shot Speaker Adaptation by Bayesian Optimization. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7639–7643, doi:10.1109/ICASSP40776.2020.9054301.

32. **Nanjo, H.** and **T. Kawahara** (2003). Unsupervised Language Model Adaptation for Lecture Speech Recognition. *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*. URL https://www.isca-speech.org/archive_open/archive_papers/sspr2003/sspr_map10.pdf.

33. **Povey, D.**, **A. Ghoshal**, **G. Boulianne**, **L. Burget**, **O. Glembek**, **N. Goel**, **M. Hannemann**, **P. Motlicek**, **Y. Qian**, **P. Schwarz**, *et al.* (2011). The Kaldi Speech Recognition Toolkit. *IEEE 2011 Workshop on Automatic Speech Rrecognition and Understanding (ASRU)*. URL http://infoscience.epfl.ch/record/192584.

34. **Prakash, A.**, **A. Leela Thomas**, **S. Umesh**, and **H. A Murthy** (2019). Building Multilingual End-to-End Speech Synthesisers for Indian Languages. *Proc. 10th ISCA Speech Synthesis Workshop (SSW)*, 194–199, doi:10.21437/SSW.2019-35.

35. **Prakash, A.** and **H. A. Murthy** (2020). Generic Indic Text-to-Speech Synthesisers with Rapid Adaptation in an End-to-End Framework. *Proc. Interspeech 2020*, 2962–2966, doi:10.21437/Interspeech.2020-2663.

36. **Prakash, A.**, **J. J. Prakash**, and **H. A. Murthy** (2016). Acoustic Analysis of Syllables Across Indian Languages. *Proc. Interspeech 2016*, 327–331, doi:10.21437/Interspeech.2016-1127.

37. **Prakash, J. J.**, **G. B. Rajan**, and **H. Murthy** (2018). Transcription Correction for Indian Languages Using Acoustic Signatures. *Proc. Interspeech 2018*, 3177–3181, doi:10.21437/Interspeech.2018-1188.

38. **Prasad, V. K.**, **T. Nagarajan**, and **H. A. Murthy** (2004). Automatic segmentation of continuous speech using minimum phase group delay functions. *Speech Communication*, **42**(3), 429–446. ISSN 0167-6393, doi:https://doi.org/10.1016/j.specom.2003.12.002.

39. **Prenger, R.**, **R. Valle**, and **B. Catanzaro** (2019). Waveglow: A Flow-based Generative Network for Speech Synthesis. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3617–3621, doi:10.1109/ICASSP.2019.8683143.

40. **Raj, D.**, **D. Snyder**, **D. Povey**, and **S. Khudanpur** (2019). Probing the Information Encoded in X-Vectors. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 726–733, doi:10.1109/ASRU46091.2019.9003979.

41. **Ramani, B.**, **S. L. Christina**, **G. A. Rachel**, **V. S. Solomi**, **M. K. Nandwana**, **A. Prakash**, **S. A. Shanmugam**, **R. Krishnan**, **S. K. Prahalad**, **K. Samudravijaya**, **P. Vijayalakshmi**, **T. Nagarajan**, and **H. A. Murthy** (2013). A Common Attribute based Unified HTS framework for Speech Synthesis in Indian languages. *Proc. 8th ISCA Workshop on Speech Synthesis (SSW 8)*, 291–296. URL https://www.isca-speech.org/archive_v0/ssw8/papers/ssw8_291.pdf.

42. **Rangarajan, V.** and **S. Narayanan** (2006). Analysis of disfluent repetitions in spontaneous speech recognition. *2006 14th European Signal Processing Conference (EUSIPCO)*, 1–5. URL https://ieeexplore.ieee.org/abstract/document/7071351.

43. **Sainburg, T.** (2019). timsainb/noisereduce: v1.0. *Zenodo*, doi:10.5281/zenodo.3243139.

44. **Sainburg, T.**, **M. Thielk**, and **T. Q. Gentner** (2020). Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires. *PLOS Computational Biology*, **16**(10), 1–48, doi:10.1371/journal.pcbi.1008228.

45. **Sen, A.** and **K. Samudravijaya** (2002). Indian accent text-to-speech system for web browsing. *Sadhana*, **27**(1), 113–126, doi:10.1007/BF02703316.

46. **Shanmugam, S. A.** (2015). A hybrid approach to segmentation of speech using signal processing cues and hidden Markov models. *MS Thesis, Department of Computer Science Engineering, IIT Madras, India*. URL https://sas91.github.io/pdf/Aswin-MS_thesis_IITM.pdf.

47. **Shen, J.**, **R. Pang**, **R. J. Weiss**, **M. Schuster**, **N. Jaitly**, **Z. Yang**, **Z. Chen**, **Y. Zhang**, **Y. Wang**, **R. Skerrv-Ryan**, **R. A. Saurous**, **Y. Agiomvrgiannakis**, and **Y. Wu** (2018). Natural TTS Synthesis by Conditioning Wavenet on Mel Spectrogram Predictions. *ICASSP 2018 - 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4779–4783, doi:10.1109/ICASSP.2018.8461368.

48. **Sisman, B.**, **M. Zhang**, **M. Dong**, and **H. Li** (2019). On the Study of Generative Adversarial Networks for Cross-Lingual Voice Conversion. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 144–151, doi:10.1109/ASRU46091.2019.9003939.

49. **Snyder, D.**, **D. Garcia-Romero**, **G. Sell**, **D. Povey**, and **S. Khudanpur** (2018). X-Vectors: Robust DNN Embeddings for Speaker Recognition. *ICASSP 2018 - 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5329–5333, doi:10.1109/ICASSP.2018.8461375.

50. **Stylianou, Y.**, **O. Cappe**, and **E. Moulines** (1998). Continuous probabilistic transform for voice conversion. *IEEE Transactions on Speech and Audio Processing*, **6**(2), 131–142, doi:10.1109/89.661472.

51. **Sundaram, S.** and **S. Narayanan** (2003). An Empirical Text Transformation Method for Spontaneous Speech Synthesizers. *Eighth European Conference on Speech Communication and Technology (EUROSPEECH 2003)*. URL https://www.isca-speech.org/archive_v0/archive_papers/eurospeech_2003/e03_1221.pdf.

52. **Székely, É.**, **G. E. Henter**, **J. Beskow**, and **J. Gustafson** (2019*a*). Spontaneous Conversational Speech Synthesis from Found Data. *Proc. Interspeech 2019*, 4435–4439, doi:10.21437/Interspeech.2019-2836.

53. **Székely, É.**, **G. E. Henter**, and **J. Gustafson** (2019*b*). Casting to corpus: Segmenting and Selecting Spontaneous Dialogue for TTS with a CNN-LSTM Speaker-dependent Breath Detector. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6925–6929, doi:10.1109/ICASSP.2019.8683846.

54. **Székely, E.**, **J. Mendelson**, and **J. Gustafson** (2017). Synthesising Uncertainty: The Interplay of Vocal Effort and Hesitation Disfluencies. *Proc. Interspeech 2017*, 804–808, doi:10.21437/Interspeech.2017-1507.

55. **Thomas, A. L.**, **A. Prakash**, **A. Baby**, and **H. A. Murthy** (2018). Code-switching in Indic Speech Synthesisers. *Proc. Interspeech 2018*, 1948–1952, doi:10.21437/Interspeech.2018-1178.

56. **Toda, T.**, **A. Black**, and **K. Tokuda** (2005). Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter. *ICASSP 2005 - 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, I_9–I_12 Vol. 1, doi:10.1109/ICASSP.2005.1415037.

57. **Toda, T.**, **H. Saruwatari**, and **K. Shikano** (2001). Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum. *ICASSP 2001 - 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (ICASSP)*, **2**, 841–844, doi:10.1109/ICASSP.2001.941046.

58. **Tokuda, K.**, **H. Zen**, and **A. W. Black** (2002). An HMM-based Speech Synthesis System applied to English. *Proceedings of 2002 IEEE Speech Synthesis Workshop (SSW)*, 227–230, doi:10.1109/WSS.2002.1224415.

59. **Valbret, H.**, **E. Moulines**, and **J. Tubach** (1992). Voice transformation using PSOLA technique. *Speech Communication, EUROSPEECH 1991*, **11**(2), 175–187, doi:10.1016/0167-6393(92)90012-V.

60. **Viswanathan, M.** and **M. Viswanathan** (2005). Measuring speech quality for text-to-speech systems: Development and assessment of a modified mean opinion score (MOS) scale. *Computer Speech and Language*, 55–83 Vol–19, doi:10.1016/j.csl.2003.12.001.

61. **Wang, Y.**, **R. Skerry-Ryan**, **D. Stanton**, **Y. Wu**, **R. J. Weiss**, **N. Jaitly**, **Z. Yang**, **Y. Xiao**, **Z. Chen**, **S. Bengio**, **Q. Le**, **Y. Agiomyrgiannakis**, **R. Clark**, and **R. A. Saurous** (2017). Tacotron: Towards End-to-End Speech Synthesis. *Proc. Interspeech 2017*, 4006–4010, doi:10.21437/Interspeech.2017-1452.

62. **Watanabe, S.**, **T. Hori**, **S. Karita**, **T. Hayashi**, **J. Nishitoba**, **Y. Unno**, **N. Enrique Yalta Soplin**, **J. Heymann**, **M. Wiesner**, **N. Chen**, **A. Renduchintala**, and **T. Ochiai** (2018). ESPnet: End-to-End Speech Processing Toolkit. *Proc. Interspeech 2018*, 2207–2211, doi:10.21437/Interspeech.2018-1456.

63. **Yamagishi, J.**, **T. Nose**, **H. Zen**, **Z.-H. Ling**, **T. Toda**, **K. Tokuda**, **S. King**, and **S. Renals** (2009). Robust Speaker-Adaptive HMM-Based Text-to-Speech Synthesis. *IEEE Transactions on Audio, Speech, and Language Processing*, **17**(6), 1208–1230, doi:10.1109/TASL.2009.2016394.

64. **Yan, Y.**, **X. Tan**, **B. Li**, **T. Qin**, **S. Zhao**, **Y. Shen**, and **T.-Y. Liu** (2021). Adaspeech 2: Adaptive Text to Speech with Untranscribed Data. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6613–6617, doi:10.1109/ICASSP39728.2021.9414872.

65. **Ze, H.**, **A. Senior**, and **M. Schuster** (2013). Statistical parametric speech synthesis using deep neural networks. *ICASSP 2013 - 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7962–7966, doi:10.1109/ICASSP.2013.6639215.

66. **Zhao, S.**, **T. H. Nguyen**, **H. Wang**, and **B. Ma** (2020). Towards Natural Bilingual and Code-Switched Speech Synthesis Based on Mix of Monolingual Recordings and Cross-Lingual Voice Conversion. *Proc. Interspeech 2020*, 2927–2931, doi:10.21437/Interspeech.2020-1163.

67. **Zhou, Y.**, **X. Tian**, **H. Xu**, **R. K. Das**, and **H. Li** (2019). Cross-lingual Voice Conversion with Bilingual Phonetic Posteriorgram and Average Modeling. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6790–6794, doi:10.1109/ICASSP.2019.8683746.

68. **Zhu, C.** and **Y. Yu** (2012). Voice conversion with UBM and speaker-specific model adaptation. *2012 IEEE 11th International Conference on Signal Processing*, **1**, 553–556, doi:10.1109/ICoSP.2012.6491548.

CURRICULUM VITAE

1. NAME : Bhagyashree Mukherjee

2. DATE OF BIRTH : 06 April, 1996

3. EDUCATIONAL QUALIFICATIONS

2018 Bachelor of Technology (B. Tech.)

Institution : Institute of Engineering & Management, Kolkata

Specialization : Department of Computer Science & Engineering

Master of Science (M. S. by Research)

Institution : Indian Institute of Technology Madras

Specialization : Department of Computer Science & Engineering

Registration Date : 01 January 2020

GENERAL TEST COMMITTEE

CHAIRPERSON	:	Dr. Krishna Moorthy Sivalingam Professor Department of Computer Science & Engineering
GUIDE	:	Dr. Hema A Murthy Professor Department of Computer Science & Engineering
MEMBERS	:	Dr. Umesh S Professor Department of Electrical Engineering
		Dr. Arun Rajkumar Assistant Professor Department of Computer Science & Engineering