Bachelor of Science in Computer Science & Engineering

Development of a Bangla Speech to Text Conversion System Using Deep Learning

by

Srijoni Saha

ID: 1504091

Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)

Chattogram-4349, Bangladesh.

April, 2021

Development of a Bangla Speech to Text Conversion System Using Deep Learning

Submitted in partial fulfilment of the requirements for
Degree of Bachelor of Science
in Computer Science & Engineering

by

Srijoni Saha
ID: 1504091

Supervised by

Prof. Dr. Asaduzzaman
Professor
Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)
Chattogram-4349, Bangladesh.
April, 2021

The thesis titled '**Development of a Bangla Speech to Text Conversion System Using Deep Learning '** submitted by ID: 1504091, Session 2019-2020 has been accepted as satisfactory in fulfilment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering to be awarded by the Chittagong University of Engineering & Technology (CUET).

Board of Examiners

______________________________________ Chairman

Prof. Dr. Asaduzzaman
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

______________________________________ Member (Ex-Officio)

Prof. Dr. Asaduzzaman
Professor & Head
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

______________________________________ Member (External)

Prof. Dr. Md. Mokammel Haque
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Declaration of Originality

This is to certify that I am the sole author of this thesis and that no part of it, nor the whole thesis, has been submitted to any other institution for a degree.

To the best of our understanding, this research does not infringe on anyone's copyright or breach any proprietary rights, and that any inventions, methods, quotations, or other content from other people's work used in this thesis, whether published or not, is completely accepted in compliance with normal referencing practices. I'm also aware that if any copyright infringement is discovered, whether deliberate or not, I can face legal and disciplinary action from the CUET Department of CSE.

I hereby grant all rights in the copyright of this thesis work to the Department of CSE, CUET, who shall be the owner of the copyright of this work, and any duplication or use in any form or by any means without the permission of the Department of CSE, CUET is prohibited.

Signature of the candidate

Date:

Acknowledgements

I am appreciative to Almighty God who has enabled me to finish my venture and to mean myself in playing out the consummation of B.Sc. Engineering degree. I am obliged to my supervisor Prof. Dr. Asaduzzaman, Professor, Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, for his support, genuine direction, constructive criticisms and unending tolerance all through the advancement of the task. All these things contributed to humble advancement as a network analysis practitioner. He upheld me by giving books, journal papers, necessary hardware and significant exhortation. From absolute starting point sir consistently empowered me with legitimate rule so the undertaking never appeared to be a weight to me. I might want to stretch out my appreciation to the entirety of my instructors for their important direction in each progression of mine during the four years of learning stage. I might likewise want to thank my family for their steady love, blessings and backing. At long last, I might want to thank my companions, senior, youngsters and the staffs, who straightforwardly or in a roundabout way have loaned their hand in this endeavor.

Abstract

Bangla Speech-To-Text (STT) conversion is a technology that provides a means of converting spoken Bangla language to a written Bangla text form. The standard of speech recognition in different languages is rising step by step however Bangla speech recognition has drawn an exceptionally little attention. Building up a STT framework is a bulky procedure and it requires a few stages, for example, extracting features from raw audio, understanding relation between audio features and phonemes, converting phonemes into actual spelling with pronunciation model and so on. Deep neural network based architecture replaces these stages with neural network components which makes the task simpler and removes dependency in hand engineered rules. In this undertaking, an implementation of DeepSpeech2 architecture is proposed for Bangla STT system which generates text from speech using neural networks trained using only speech examples and corresponding text transcripts. The architecture is comprised of many recurrent layers, convolution layers and a fully connected layer. This implementation shows that it can accommodate a wide range of speech samples recorded by people of various ages and genders. This report shows a comparison between genuine text transcript with produced text transcript for a similar sound example. Comparison of results between implemented architecture and already existing Bangla STT has also been presented in this report. The report is finished up with a discussion about the word error rate and implementation challenges.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The process of recognizing speech and creating a text transcript for it is known as speech to text conversion. Since speech is such an intuitive mode of communication, this technology has the potential to have far-reaching effects in easing human-machine interaction. Despite the fact that a lot of work involving various algorithms has been done in the field of speech recognition in general, very little has been done in the field of speech recognition in Bangla. This report details a deep learning-based speech-to-text conversion method for the Bangla language.

1.2 Speech to Text Conversion

Speech is likely the most effective approach to communicate with one another. On the off chance that human-PC correspondence is delivered conceivable through speech, at that point human will have the option to associate with machine in the most natural manner. PC interfaces are included a keyboard and mouse. Human needs to get capable in specific aptitudes before they use PC. Speech to text system ease this communication barricade by making the computer able to recognize the human's instinctive communication technique. A speech-to-text (STT) system generates methodologies and technologies that empowers the interpretation of communicated in language into text.

An automatic speech recognition (ASR) system may be a hardware and software system, where the input is the sound of the voice (speech) and the output is the identification of those spoken words. The fundamental objective of an automatic speech recognition system (ASR) is to construct a system that will ”understand” our oral communication and react — this implies the system will react suitably to the verbally expressed words and convert the speech into another medium like text.

Speech-to-text model has been advantageous for vision weakened people from the earliest starting point. It has an expected advantage for understudies with physical inabilities and extreme learning handicaps. Speech recognizer or Speech-to-text system usages are constrained to the individuals with inabilities as well as now they are utilized for individual use on regular routine. Individuals can give spoken directions to the machine through this framework.

The positioning of Bangla is seventh among 100 spoken languages across the wold, having around 256 million native and non-native speakers [1]. The standard of speech recognition in different languages is rising step by step however Bangla speech recognition has drawn an exceptionally little attention [2]. STT framework for Bangla language can extraordinarily help the human PC association: the conceivable outcomes are inestimable such a framework can help conquer the education hindrance of the basic masses, engage the outwardly disabled populace, increment the potential outcomes of improved man-machine interaction, automation of translation, composing books/texts utilizing your own sound just, empowering complicated analysis on information using the generated textual files and a great deal of different things. Truth be told, communicating with technological devices via voice has become so popular.

1.2.1 Objectives

1. To correct the mistranscriptions of the ASR data set and prepare the data set for training.

2. To implement a Speech to Text conversion system for Bangla using deep learning based architecture DeepSpeech2 which will generate text directly from audio.

3. To evaluate the performance of the proposed technology in real world environment and comparing the result with Google Bangla STT API.

1.3 Framework Overview

Traditional approach of STT shown in Fig.1.1 needs multiple stages of data processing. The concept of phoneme is also necessary in traditional pipeline. Through deep learning End-To-End speech recognition system shown in Fig.1.2 can be built which can replace all the multiple stages to a single neural network. It will be possible to recognize speech without phoneme dictionary, even without the concept of a ”phoneme”. For End-To-End learning there are 2 challenges. Building large labeled training set and training networks that are large enough to effectively utilize all of these data [3].

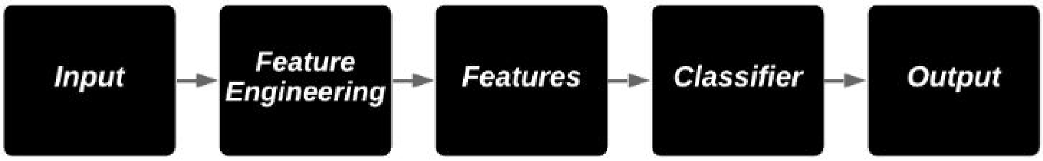


Figure 1.1: Traditional Speech Recognition System

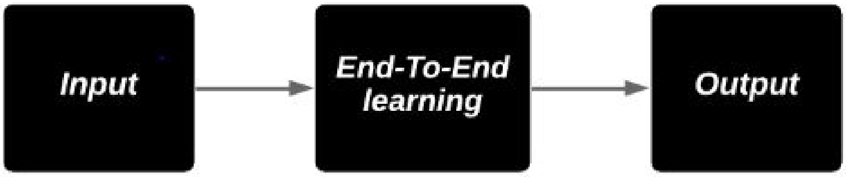


Figure 1.2: End-To-End Speech Recognition System

1.4 Difficulties

In other languages, the quality of speech recognition is improving all the time, but Bengali speech recognition has gotten very little attention. This is most likely due to the complexity of Bengali words and the fact that the total hour of bangla speech data set is insufficient to produce the best results. Working with speech data presents several obstacles, such as massive quantities of data are needed to construct the speech recognition system. Audio files in the dataset have to be collected from people of various genders and ages. Processing these audio to make it appropriate for training is an important part of the project. It takes a long time to process a huge amount of audio data. Another challenge is speeches vary in duration. The same word can be spoken for different lengths of time [4].

1.5 Applications of STT

Probably the most useful application field of speech recognition is the writing and communication aids for the blind. It's easier to interact with computer with speech rather than from Braille characters. Using speech recognition system in mobile phone, they will be able to make phone calls, send text messages with their voice.

Speech recognition is used widely as a digital assistant. For example, searching for necessary document on computer through speech. Documents can be generated faster with speech recognition as they can if they are typed. In medical treatment where seconds are pivotal, hand-free & instantaneous access to information can have a substantial positive impact on patient welfare and medical efficacy. For example, quickly finding information from medical records through speech.

Passwords are sequence of characters which a machine takes to verify the intended user. But if voice is used as password it will be safer because different person's voice has different exceptional characteristics. With this exceptions, a person's voice could a reliable approach to verify a user [5].

1.6 Motivation

The previous approaches of speech-to-text conversion are comprised of several components. Decoder represents words as a sequence of "phonemes". Pronunciation model converts the transcriptions in phonemes into actual spelling. This adds more complexity to the system. Acoustic model associates audio features with phonemes. This traditional pipeline hard to get working well, like for different accent this is problematic. Each part of system has it's own challenges. Because of all these components building a STT system has always been a cumbersome process for a long time. The result of these STT system was also not so good. In recent years with growing application of machine learning and deep learning many architectures have shown impressive result for a STT system. Some architectures have successfully generated text which is impressive. DeepSpeech2 a deep learning based architecture for STT system suggested by Baidu claimed that this architecture is able to recognize human speech. DeepSpeech2 has been successfully implemented for 2 languages - English & Mandarin. We intend to implement this architecture for Bangla language and develop a deep learning STT system for Bangla.

1.7 Contribution of the Thesis

The main objective of this work is to develop a STT conversion system for Bangla language using deep learning which will generate Bangla text from a given Bangla speech. According to our work this STT will be implemented using Baidu's DeepSpeech2 architecture which is able to generate text with only trained on <audio,text> pair. This implementation promises an end-to-end speech-to-text system.

The key objectives of this work may mention in the following:

- To implement a speech-to-text conversion system for Bangla using deep learning-based architecture DeepSpeech2 which will generate text directly from audio.
- To evaluate, observe and analyze the performance of the proposed system.
- To compare the performance of the proposed innovative technology in real environment with existing works.

1.8 Thesis Organization

The remainder of the report is structured as follows. In the next chapter, an overview of our project related terminologies, the popular methods for speech to text conversion systems with their limitations and also existing Bangla speech to text systems are presented. Chapter 3 describes the components and working procedure of DeepSpeech2 [3] architecture. In Chapter 4, we have illustrated our implementation of the project in details. Chapter 5 centers on the experimental result of the proposed system. In order to evaluate the system, we have used both training and testing data and also comparison with existing Bangla STT has been discussed with respect to word error rate. The project concludes with future plan of our work in Chapter 6.

1.9 Conclusion

This chapter provides a description of the Bangla STT. This chapter discusses the challenges and provides a quick overview of the speech to text system. The inspiration for this work, as well as the contributions made, are also listed here. The context and current state of the problem will be discussed in the following chapter.

Chapter 2

Literature Review

2.1 Introduction

One of the most significant research areas has been speech recognition. So, while researching for thesis, we came across some research works from which we gathered information to prepare for this thesis, and we have mentioned some of them here with descriptions.

2.2 Speech Recognition Basics

Speech recognition is a wide term which implies it can perceive nearly anyone's speech but big amount of training data is required.

2.2.1 Types of Speech

There are different kinds of SR on basis of what kind of speech they can recognize [6]

- Isolated speech recognition
- Connected speech recognition
- Continuous speech recognition

In an isolated-word speech recognition system the speaker need to pause briefly on both side of the utterance, whereas a continuous speech recognition system does not. Connected word speech recognition is similar to isolated-word speech recognition but the pause is minimum.

2.2.2 Types of Speaker model

- 1. Speaker independent model
- 2. Speaker dependent model

The system which is dependent on speaker, are mapped around a particular speaker. Speaker independent systems are mapped for different speakers [6].

To comprehend SR technology, some of the terms that are utilized all through the full report and had to know are :

- Phonemes : These are units of sounds that we pronounce as we speak.
- Grapheme : Characters in a text, smallest unit of writing system for any language.
- Utterance : Can be a word, a few words, a sentence or even multiple sentences.
- RNN : Recurrent neural network is a type of neural network where the output from previous step are stored and fed as input for the current step. In traditional neural networks, inputs and outputs are independent of each other, but there are case for example, if to predict the next word of a sentence, the previous words are necessary. So we can use RNN.
- CNN : Convolutional neural network is a type of neural network in which feature extraction is done using convolution and polling. In this network kernels are feature detectors. For audio we use spectograms. We can visualize the audio as an image. CNN can be used for extracting features of audio.
- Softmax layer : The final output layer that computes probability distribution over characters.
- Batch normalization : A technique that normalizes activations of particular layer in a network. It increases the training speed.
- CTC loss function : Connectionist temporal classification loss function is a technique for mapping variable length output to fixed length transcript.
- Sampling rate : The average no of samples obtained in one second.

- Acoustic model : Tries to understand the relation between audio features and words.
- LSTM : Long short-term memory is an artificial recurrent neural network (RNN). It's major characteristics is - it not only process single data points, but also can process entire sequences of data (such as speech or video) that's why LSTM is applicable for speech recognition.
- GRU : Gated recurrent units are like LSTM with a forget gate but has fewer parameters than LSTM [3].

2.3 Background and Present State of the Problem

Work into automated speech recognition has been performed for nearly four decades. The earliest attempts to devise machine-based automatic speech recognition systems were made in the 1950s, when different researchers attempted to exploit the basic ideas of acoustic phonetics [7]

In 1962 IBM made a digit recognizer named ”SHOEBOX.” It could acknowledge connected digit strings. It had sixteen words in its vocabulary. It was an isolated word speech recognition for a single speaker. A phoneme recognizer was built in 1959 to acknowledge nine consonants and four vowels. For making recognition decisions, a spectrum analyzer and a pattern matcher were used. But the great thing about the research was to use statistical information about permissible phonemic sequences for words consisting of two or more phonemes.The main emphasis of the 1970's was isolated word recognition [7]. Until the 1970's and 1980's ASR was mainly known as a speech analysis technique. But that approach was only good for identifying speakers, not for recognizing speech.

2.3.1 Statistical Parametric Based Recognition

This approach recognizes speech by integrating parameters such as fundamental frequency, the lowest frequency of a periodic waveform that defines the sound pitch and processes it to produce text. There are two stages to a parametric STT system.

First of all, it would be to extract features such as cepstral, spectrogram, fundamental frequency, etc. that reflect some inherent characteristics of human speech. For example, Cepstra is about approximating the human speech transfer function. That means these features are hand engineered.

Second, linguistic features such as phonemes will be used for text generating.

2.3.1.1 Advantage

- For storing recorded units, take less memory.
- It works for database of multiple speakers.

2.3.1.2 Disadvantage

- Have problems with data alignment [8].

2.3.2 Speech recognition using Deep learning

Models of Deep Learning have proven exceptionally efficient in learning inherent data features. These features aren't really human readable, but they're computer-readable and much better for a model to represent data. The more data goes forward into a neural network, the more complicated features are identified. This is another way of saying that a model of Deep Learning learns a function to map input X to output Y.

Working on this assumption now, Speech-to-Text should have input X as an audio waveform and output Y as a text string. It should not use any hand engineered features, but rather learn to represent human speech in text format with new high dimensional features. That is what we are trying to accomplish.

Baidu Research Center published it's research named 'DeepSpeech2,' a neural network implementation of Speech-to-Text conversion. In a nutshell, it works like this: to map a sequence of audio features to a sequence of letters, we use a sequence-to-sequence model optimized for STT. It provides a set of novel techniques for data synthesis that allow a large amount of varied data to be efficiently obtained for training. The output layer calculates CTC loss function to measure the prediction error [9].

2.3.2.1 Advantage

- Can understand those data where order is so important.
- Can generate text with only using <audio,text> pair for training.

2.3.2.2 Disadvantage

- Takes a lot of time for training.
- Requires better GPU configuration.
- Needs a huge amount of data.

2.4 Related Literature Review

- For recognizing speech Nidhi et al. [10] utilized Neural System (NN) along with Mel frequency Cepstrum Coefficients (MFCC). Mel frequency Cepstrum Coefficients (MFCC) has been utilized for the feature extraction of audio. The feature of the waveform can be achieved by this. Feed-forward neural network with back propagation algorithm has been applied for pattern matching. MFCC is progressively favored in feature extraction as it produces the training vectors by transforming speech signal into frequency domain. Consequently it is less prone to noise. Likewise this methodology is employed as an outcome of human hearing relies on frequency analysis. MFCC mimics the human's ear's conduct. But this paper worked with exceptionally little data set. The voice of two persons-one male and one female with just two words was recorded.
- Neural network acoustic models and other connectionist approaches were first introduced to speech pipelines in the early 1990s [8]. Hidden markov models are statistical method of speech recognition. This model could generalize better. The core acoustic pattern matching of speech recognition is two types. Local and global matching. Representations of speech frames are compared to spectra of speech that were used for training, using some measure of similarity or distance. Each of these comparison can be viewed as local match. The global match is the search for best sequence of words (in the sense of the best match to the data).

Language model re-scores the sentences according to grammar and semantics. But HMM model uses specially constructed language model. HMM has data alignment problem.

- The system developed by Abul et al. [10] uses HMM technique for recognizing pattern and language model. During signal processing noise was removed from audio sample for better result. But only 100 words were used.
- A Bangla STT system using HMM with MFCC features was represented by Ghulam et al. [11] . This is actually a digit recognizer. But the system fails for recognizing digits with similar pronunciation.
- Adnan et al. [12] proposed a Bangla STT conversion system using spectogram and fuzzy logic. Using fuzzy learning methodology the computer-system was trained. Short-time fourier transform (STFT) was used for generating spectogram. But the data set contains only 50 words. This was just an isolated word speech recognition system.
- Ali et al. [13] used ANN with back propagation for Bangla speech recognition. But only 300 audio samples were used containing only 10 digits. The audio samples in data set were recorded in noise free environment. This was just an isolated word recognition system.
- The framework created by Shikhor et al. [14] takes voice as input from clients and afterward the artificial neural network is trained using feed forward back propagation algorithm which maps this Bengali voice into text format. The input sound is taken as variable. Variable stores audio records as double valued vector matrix. At that point preprocessing of vector matrix is finished. The double valued vector matrix is changed over to integer valued vector matrix. A gape remains before and after the original audio signal which is really the preparation for delivery. Disposing of that signal converts the vector matrix to sample input matrix. Consonant is included vowel and consonant. Vowel part is disposed of. At that point acknowledgment is finished. Limitations of this paper is this worked with Bengali characters only.

- Prachi et al. [6] gave summary of major technological point of view and gratefulness of the basic progress of conversion from speech to text and additionally gave outline strategy created in every stage of classification of speech to text conversion. Isolated word acknowledgment is done here. In preparation stage digit is recorded utilizing PCM, saved as wave document utilizing software of sound recording. Wavered command of MATLAB converts wave file to speech samples. Through voice action identification speech estranged from pauses. Using linear predictive coding speech analysis and synthesis is finished. For each word in the vocabulary, framework assembles a HMM model and trains the model during training stage. In recognizing stage unknown input pattern is recognized by thinking about references. Limitations of this paper is this worked with just 10 digits(0-9) as vocabulary.

- Speech Application Program Interface (SAPI) is developed by Microsoft Corporation for speech related works in its Windows operating systems including features for only eight languages. English is one of that languages. Shaheena et al. [15] aimed of to investigate Speech-to-Text (STT) conversion using SAPI for Bangla language. XML grammar file is generated for using SAPI. XML grammar is compiled to binary grammar format by using SAPI CFG compiler. Binary grammar loaded into SAPI. If Bangla word is spoken, SAPI search it in the binary grammar file. Corresponding pronunciation of Bangla word is returned with English character when match occurs. But problem is this is a slow process and its operations are sequential. Bangla speech is recognized word by word basis. A person should use proper break in each word during speaking so that system writes the word if match occurs. The output text is in English characters.

- Pialy et al. [16] represented the result of a preliminary study to recognize the human speech using mel-frequency cepstrum coefficients (MFCC) features. Neural Network is trained using the MFCC features of 2 speakers. Only 1 specific person will be acknowledged with his command and the system is terminated for the different. The result of matching features in a neural network shows that MFCC features work significantly toward speech recognition. But during this research only two sentences “BATI JALAW”, ”PAKHA BONDHO KORO” were used in the data set.

- Mahadi et al. [4] presented a speech recognition system using double layered LSTM-RNN approach. Each word was separated into a no of frames which contain 13 MFCCs. The model's final layer has a softmax layer which has equal no of units to the no of phonemes. Most probable phoneme was picked for every time step. But the research work is in word level and used a very small data set having 2000 words. Bangla speech to text conversion system was implemented focusing on only Bangla real number audio data set.
- Tahsin et al. [17] constructed a Bangla STT with deep recurrent neural network. In this paper, Deep Speech [9] neural network architecture was implemented for Bangla language. The network can find out the best possible way for mapping frames of a sequence of audio to characters through training with a large data set. The prediction was done with RNN and a language model. But the proposed system achieved 50 percent accuracy on testing data. The data set was too small having only 1400 sound clips.
- Sakhawat et al. [18] proposed an end to end speech recognition for Bangla. Sequence to sequence model was used here. Here RNN was used with CTC criterion. Spectogram feature sequences from raw audio sample is used in this proposed model. A data set of 350 hr is used. Character error rate (CER) is less. But the result of word error rate (WER) is not good.

2.5 Deep Speech2 model

Dario et al. [3] represented a progressive speech recognition system developed by end-to-end deep learning. This design is considerably less complicated than ancient speech systems that uses hand engineered process pipelines. The network is comprised of many layers of recurrent connections, convolution filters and batch normalization applied on RNNs. It provides set of novel data synthesis techniques that allows to efficiently obtain a large amount of varied data for training. The output layer computes CTC loss to measure the error in prediction. Feed forward neural networks take fixed size input and gives fixed size output. Does not capture information from sequences/time series information. But RNN does this. Data alignment problem of HMM is solved by CTC. The speech system always gets improved with the number of labeled data used for training. This model was trained on 11940 hours of speech. This system can be quickly applied to a new language because this approach is so generic. It could recognize 2 very different languages, English and Mandarin. Speech of different accents and noisy environment can be converted to text with this architecture.

2.6 Conclusion

This chapter contains an useful description of the literature review. The bangla speech recognition system's past and current state are also discussed. The technique for converting Bangla speech to text using deep learning is explained in the following chapter.

2.6.1 Implementation Challenges

A large amount of labelled speech data is required to create a voice-independent speech recognition system. Audio samples collected from people of various genders and ages should be included in the data set. There is no open source standard data set that meets the needs of this study. Processing a large volume of audio data takes a long time and requires a lot of computing power and memory. It takes a long time to train a deep neural architecture with a large amount of data. As a result, maintaining a constant power supply during the training procedure is another problem.

Chapter 3

Methodology

3.1 Introduction

Thousands of hours of labelled speech cannot be exploited by a simplistic multi-layer model with a single recurrent layer. To learn from such large datasets, we used the Deepspeech2 architecture for Bangla Language, which has a large model capacity. Preparing the Bangla ASR dataset to meet the DeepSpeech2 architecture's requirements, as well as the method of implementing DeepSpeech2 architecture for Bangla language is discussed in this chapter.

3.2 Representation of the System Model

Using traditional methods, speech recognition systems required a lot of underlying steps for example, extracting features from raw audio, understanding relation between audio features and phonemes, converting phonemes into actual spelling with pronunciation model and so on. The term end-to-end denotes a system which requires less processing steps compared to similar existing system. The architecture 'DeepSpeech2' that is considered in this project is an end-to-end speech recognition system. Since this model can be trained providing only a speech database with corresponding transcripts of text.

Sufficient capacity of model is needed to work with huge amount of data for overcoming underfitting prolem. The DeepSpeech2 architecture is comprised of many bidirectional recurrent layers, convolution layers and a fully connected layer. For optimizing the model successfully Batch normalization is used with RNN. The pipeline walks through preprocessing, CTC, training and decoding [3]. The goal of acoustic model is to create a neural network from which text transcripts can be extracted. For solving the issues of acoustic model CTC criterion is used. All the components of DeepSpeech2 architecture will be discussed.

3.2.1 Convolution Layers

Convolution layers have shown very good result in image processing field. The audio can also be visualized as an image. X axis and Y axis are time and frequency. The amplitude is the pixel value. So convolution layers can extract features from audio. In fully connected layers each neuron is connected to each neuron of previous layer which makes it computationally expensive. But in convolution layers each neuron is connected to few nearby neurons of the previous layer. So improvements in automatic speech recognition task is happening with CNN based architecture [18]. In DS2 architecture convolution layers are used for preserving features of audio samples.

3.2.2 Recurrent Layers

There are lot of data where order is so important like sentence, time series. RNN understands the order of the data, can process sequential data. With RNN each data is processed on basis of context. RNN generates text sequences. At each time step, a prediction is made by RNN over characters. Time series data gives data point step at a time. So the network predicts the next step. Prediction depends on previous step, what happened before. Simple RNN does not support long term memory, can not use information from distance path, can not remember long term patterns of audio data. With more complicated recurrence allows the network remembering state over more time steps. Two special recurrent architectures are Long short term memory (LSTM) units and Gated recurrent units (GRU). GRU and LSTM have alike accuracy [3]. In this work we used LSTM.

3.2.3 Batch Normalization

Normalization is transforming the data to put all the data point in same scale. Some numerical data can be very high or very low. Features can vary very widely. Non normalized data can create instability. Weights in model get updated with each epoch. If during training one of the weight becomes drastically larger than other weights that can cause instability. With batch normalization we can normalize after activation functions of layers. Batch normalization improves convergence speed of recurrent networks and increases accuracy [3].

3.2.4 Connectionist Temporal Classification

An issue arises with acoustic model in STT system, length of the input is not equal to the length of the transcripts. Audio signals are often not well segmented in relation to transcripts and contain a bit of intrinsic noise. For performing end-to-end speech recognition, we need to align perfectly audio clips with the respective transcripts. We do not have explicit information about how audio signal aligns to the characters in the transcripts. CTC is a way to address this problem without any clear input and output sequence alignment information [18]. Given transcripts and output of the RNN are compared. CTC cost function is calculated in this architecture. RNN with CTC criterion solves data alignment issue of speech to text conversion system [3].

3.3 STT Architecture

A simplistic multi-layer model with a single recurrent layer can not exploit thousands of hours of speech with labeling. To learn well from data sets, in this architecture the potential of the model is increased by means of depth. The architecture shown in Fig.3.1 explores up to 8 layers including several bidirectional recurrent and convolution layers. It has 2 convolution layers, 5 bidirectional recurrent layers and a fully connected layer as shown in Fig.3.1. Such model has nearly 8 times the amount of computation per data example as the model in Deep Speech1 [9]. The detail of the architecture is discussed in the following.

The model is trained on <audio,text> pair. The outputs of the network are the grapheme of language. At each output time step a prediction over characters is made by recurrent network. 2 convolution layers remain in the bottom of the network. Convolution layers extract audio features. Each convolution layer uses hardtanh as activation function. Convolution network gets the feature vector then it is fed into the recurrent layers. Striding the convolution sub-sampling is done. Time series data gives data point step at a time. So LSTM makes a prediction over characters at each output time step. The

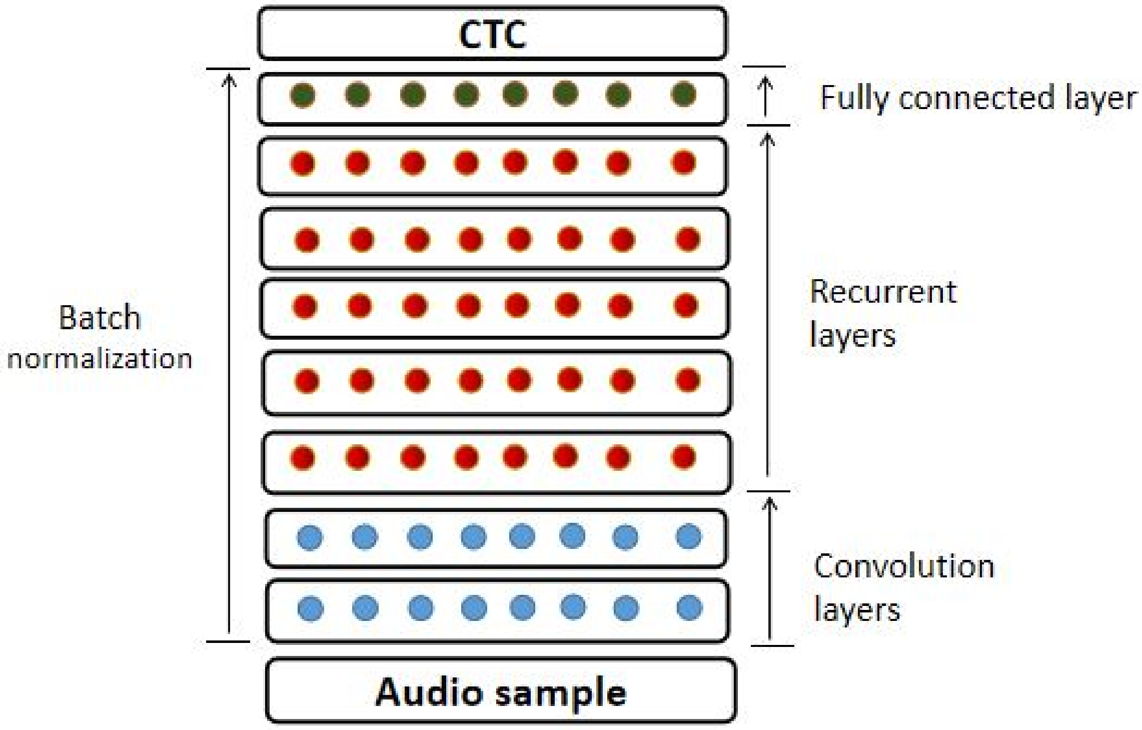


Figure 3.1: Model architecture which takes audio samples as input and gives text transcripts as output

output we get from the LSTM model is fed to the output fully connected layer. In the output softmax layer computing probability distribution over characters. One frame gives the probability of to be a specific character. The model outputs a vector that represents the entire sentence.

During training for solving the vanishing gradient problem batch normalization is used [18]. If the gradient is so small the update of weight becomes so small. The weight barely changes from it's original value, will not help to reduce the loss. The network gets stuck. The network loses the ability to learn. So batch normalization does the normalization after activation functions of layers.

The problems with audio signal is, suppose 10 sec audio has 100 samples per second. So the audio with enter with 1000 inputs but the output might not be 1000 alphabets. CTC cost function allows the recurrent layers to generate output with special blank characters.

For example if the utterance is গগন

With CTC the output can be represented with special blank symbol $: গগ$$গগন

In this way 2 characters output can be represented by 1000 output values. So LSTM with CTC criterion can solve the data alignment issue of STT system [19]. The probability of all possible alignments are checked then max probabilistic transcript is chosen as shown in fig.3.2. The CTC loss function and it's derivative with respect to the parameter of the network is calculated [3]. Then the derivative is used for updating the network parameters through the back propagation. Stochastic gradient descent method is used for optimization.

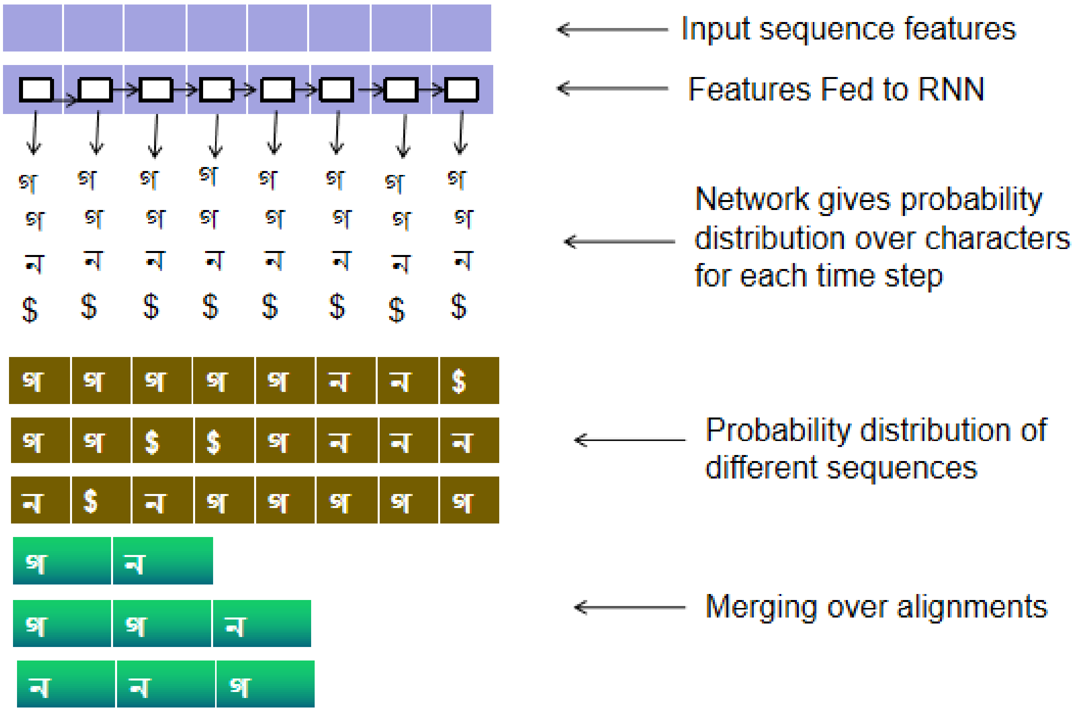

Input sequence features
Features Fed to RNN
Network gives probability distribution over characters for each time step
Probability distribution of different sequences
Merging over alignments

Figure 3.2: Text Alignment with CTC

Figure 3.3 depicts the overall workflow of this architecture's speech to text conversion. Recurrent layers understand the relationship between features and transcripts after convolution layers extract features. The features are mapped to a sequence of text transcripts by checking the probability distribution over characters at each time step and the probability distributions over the alignments.

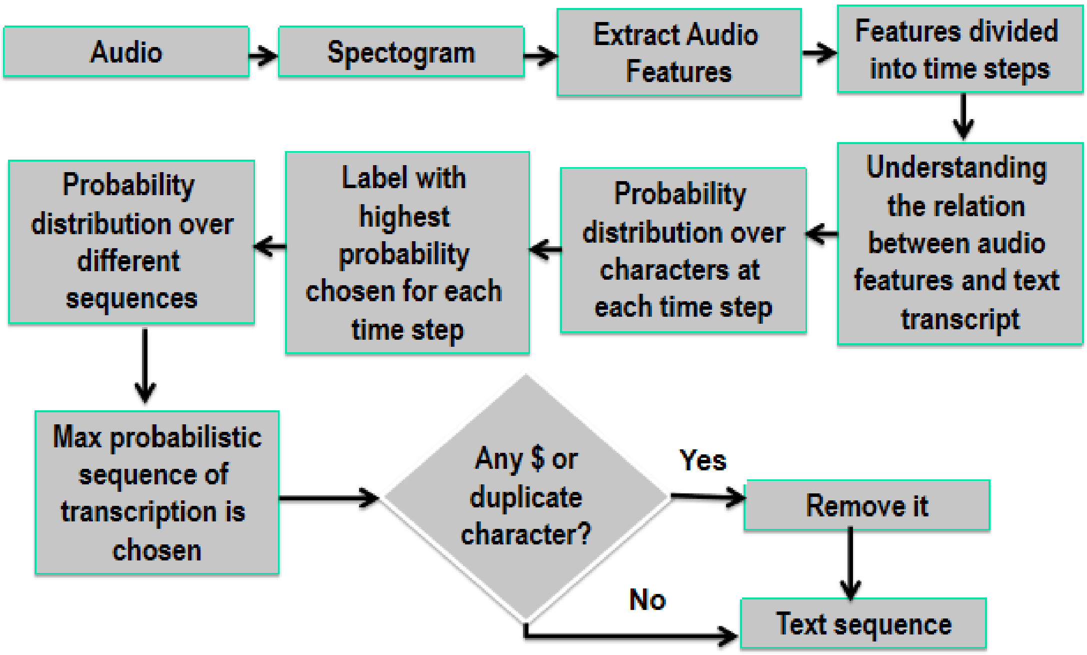

Audio
Spectogram
Extract Audio Features
Features divided into time steps
Understanding the relation between audio features and text transcript
Probability distribution over characters at each time step
Label with highest probability chosen for each time step
Probability distribution over different sequences
Max probabilistic sequence of transcription is chosen
Any $ or duplicate character?
Yes
Remove it
No
Text sequence

Figure 3.3: Workflow of the STT Architecture

3.4 Detailed Explanation

3.4.1 Implementation Details

The implementation view of this system requires implementation of architecture to fit for Bangla language and training with a Bangla speech database with corresponding transcripts. The overall model first is trained with <speech,text> pair from the training database, then with test data the model can be evaluated.

1. Hardware Specification:

   - Personal Computer

   - Core i5 processor

   - 2.50 GHz clock speed

   - CPU 8GB

   - GPU Nvidia GTX 1080 GeoForce (8GB)

2. Software Specification:

   - Operating system : Ubuntu

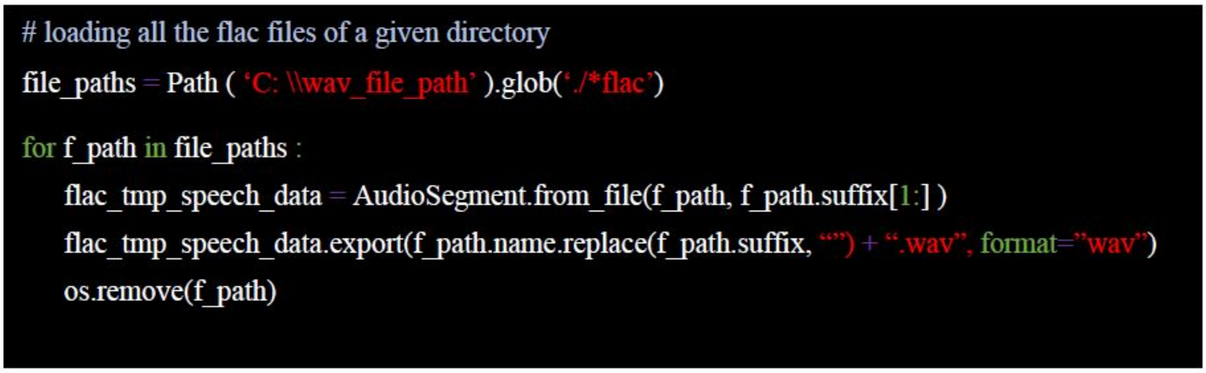
- Programming Language Used: - Python 3.6.6

3.4.2 Implementation for Bangla Language

The DS2 architecture has been implemented for English and Mandarin language. So for using this model for Bangla language the model has to be trained with Bangla speech and corresponding Bangla transcripts. Some operations were needed to prepare the Bangla data set for training.

3.4.2.1 File Format Manipulation

For training the DS2 model, all the audio files have to be in wav file format. The data set we used all the audio samples were in flac format. So we had to convert all the audio samples into wav file format using ‘Pydub’ which is a convenient audio analytics python module. The key class in Pydub named “AudioSegment” is used to modify and convert the flac file to wav file format as shown in Fig.3.4



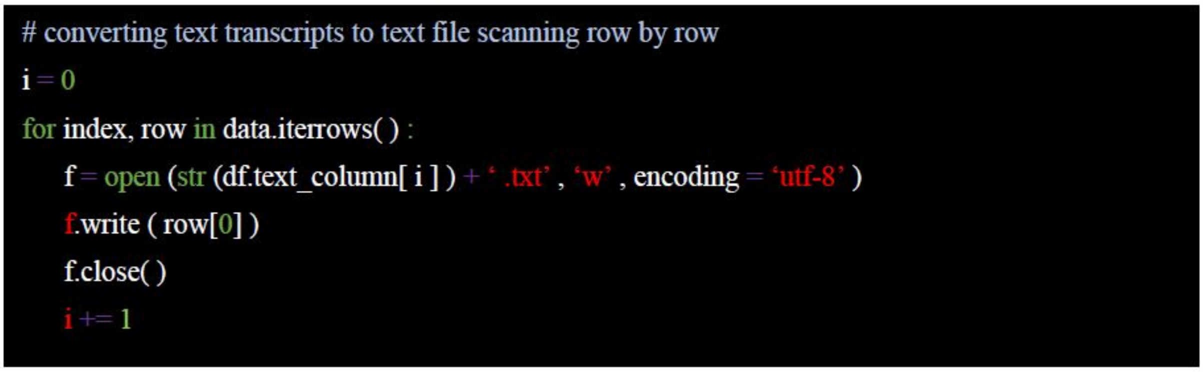
```
# loading all the flac files of a given directory
file_paths = Path ( ‘C: \\wav_file_path’ ).glob(‘./*flac’)

for f_path in file_paths :
    flac_tmp_speech_data = AudioSegment.from_file(f_path, f_path.suffix[1:] )
    flac_tmp_speech_data.export(f_path.name.replace(f_path.suffix, “”) + “.wav”, format=”wav”)
    os.remove(f_path)
```

Figure 3.4: File Format Manipulation

3.4.2.2 Text File Generation

The training pair should be an audio file and a corresponding text file. But the data set we used all the text transcripts were in single text file. But for training purpose every text transcript has to be in a text file format. So using Pandas’ “iterrows()” function we read all the text transcripts row by row from the tsv file containing all the text transcripts and converted the text transcript in each row to text file as shown in Fig.3.5. This way we generated text files for every text transcripts.



```
# converting text transcripts to text file scanning row by row
i = 0
for index, row in data.iterrows( ) :
    f = open (str (df.text_column[ i ] ) + ‘ .txt’ , ‘w’ , encoding = ‘utf-8’ )
    f.write ( row[0] )
    f.close( )
    i += 1
```

Figure 3.5: Text File Generation

3.4.2.3 Text File Organization

After generating text files for each text transcripts the we organized text files in 256 folders. The name of the folders were given as the name of the folders containing audio files.

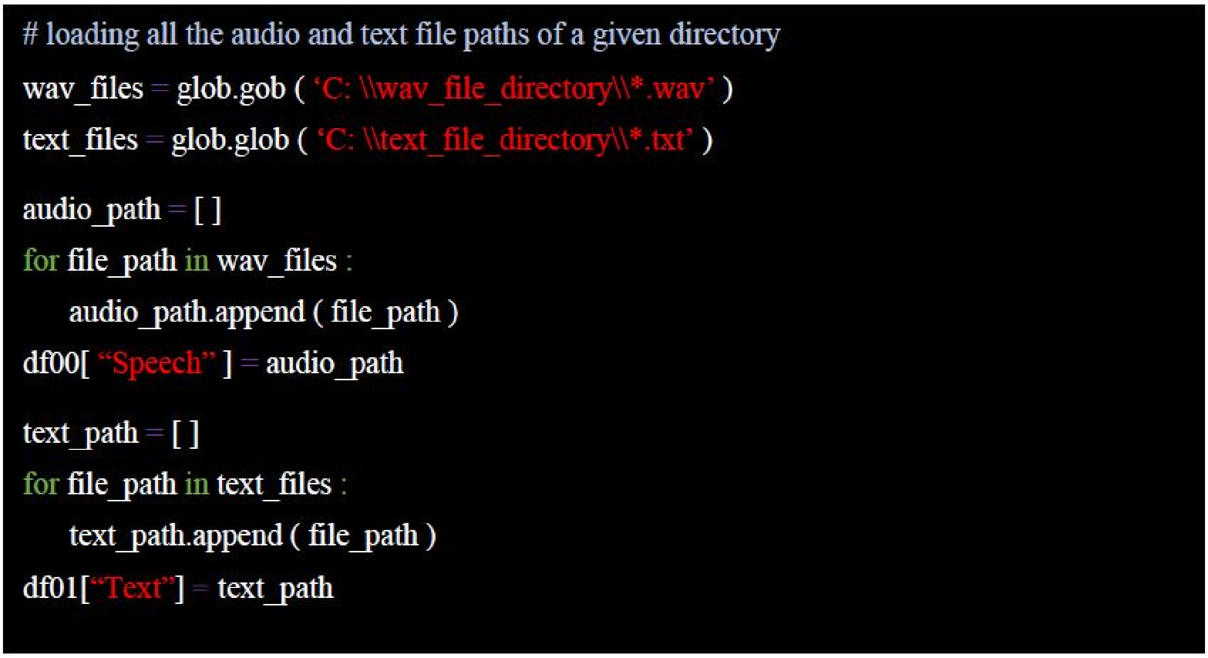
3.4.2.4 File Path Mapping

File Mapping is so necessary for working with a huge audio data. Because 218k audio files can not be kept in one single directory and track all the files from that directory. If we do that the PC will be completely unresponsive. So audio files were organized in total 256 folders. For retrieving the audio files from 256 folders for training, file path mapping was necessary.

For using custom data set with DS2 architecture a CSV file containing the locations of the training data has to be created. The format in the CSV will be like following :

/path/name.wav, /path/name.txt

For file path mapping first path has to be the file path of audio samples and the second path has to be the file path of text file containing the text transcript in one line. So as shown in Fig.3.6 using “Glob” module of python we extracted the file paths of all the audio and text files and made a csv file in which the first column contains the file paths of audio samples and the second column contains the file path of corresponding text files.



```
# loading all the audio and text file paths of a given directory
wav_files = glob.gob ( ‘C: \\wav_file_directory\\*.wav’ )
text_files = glob.glob ( ‘C: \\text_file_directory\\*.txt’ )

audio_path = [ ]
for file_path in wav_files :
    audio_path.append ( file_path )
df00[ “Speech” ] = audio_path

text_path = [ ]
for file_path in text_files :
    text_path.append ( file_path )
df01[“Text”] = text_path
```

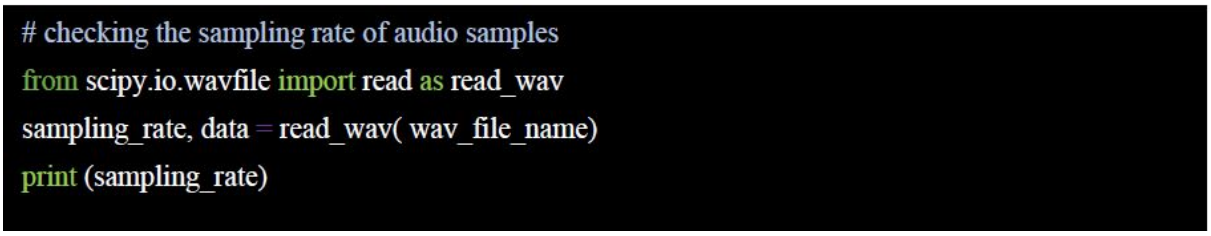
Figure 3.6: File Path Mapping

3.4.3 Correcting Mistranscriptions of Data Set

In the Google Bengali ASR data set documentation it was written that there can be corrupted transcriptions in the data set [20]. During using this data set we found that there were 218,703 audio files but text transcripts were only 127,564. So, 91,139 text transcripts were corrupted or missing. We recovered the corrupted Bangla text transcripts re-writing them and wrote the missing text transcripts hearing the audios. After all these we got a Bengali ASR data set containing 218,703 audio samples and 218,703 text transcripts without any error.

3.4.4 Audio Data Sampling Rate Checking

16kHz audio data sampling rate is perfect for speech recognition [18]. Because the bandwidth of speech is very low. 16000 samples/sec is great because it provides more precise information on high frequencies. So we checked the sampling rate of the audio samples, whether they were all 16kHz or not as shown in Fig.3.7 . If not then set the sampling rate of the audio sample to 16kHz. This way the data set was completely prepared for training.



```
# checking the sampling rate of audio samples
from scipy.io.wavfile import read as read_wav
sampling_rate, data = read_wav( wav_file_name)
print (sampling_rate)
```

Figure 3.7: Checking the Sampling Rate of Audio Samples

3.4.5 Training

The most difficult aspect of this study was the model's training. Before starting the train, we double-checked that all dependencies are installed with the correct versions and that the NVIDIA GPU driver is operational. The Cuda version must be right. Before training the model there are several hyperparameters to bear in mind. Hyperparameters are the variables that decide the structure of the network, and decide how the network is trained. Before training hyperparameters are set (before the weights and bias are being optimized). We have trained the model on 215 hours speech data with their text transcripts. The model was trained using only a batch size of 8. The size of the batch determines the amount of samples to be propagated across the network. We trained the model for 20 epochs.

3.5 Conclusion

This chapter discusses a technique for converting Bangla speech to text. Here are the various dataset preprocessing steps as well as a thorough overview of the architecture. The proposed methodology's experimental results are examined in the following sections.

Chapter 4

Results and Discussions

4.1 Introduction

This section contains the evaluation and the nature of the converted text and provides comparison between generated and actual text transcript. Comparison between results of the implemented architecture and other existing speech to text system for Bangla has been discussed in this chapter. In the discussion word error rate and challenges faced during implementation of this project has been illustrated in detailed manner.

4.2 Speech Data Set Description

One of the major step for the implementation of this architecture for Bangla was to find a long (200+ hours) speech database of different speakers with corresponding transcript for training. There was no standardized open source Bengali ASR dataset that was appropriate for this project. We found the speech corpus named "Bengali ASR dataset" published by Google for research on Bangla language processing [20] that met the requirements of this thesis. This data set includes speeches by people from a variety gender and different ages. So speaker independent speech to text conversion system can be made with this data set. But according to Google Bengali ASR data set documentation it was written that there can be corrupted transcriptions in the data set [20]. So we checked the data set and found some error. Total audio files were 218,703 but text transcripts were only 127,564. So 91,139 text transcripts were corrupted or missing. After correcting them i was able to prepare a speech data set of 215 hours and 31 minutes for this project which contains 218,703 audio samples and 218,703 corresponding text transcripts.

Brief information about the data set is given below:

- The corpus includes around 218K sentences.
- Vocabulary size : 57k words.
- Includes about 1295 unique grapheme.
- Recorded by speakers of different gender and ages.
- Includes text transcripts of the speech.
- Almost 215 hours and 31 minutes of speech.
- Audio samples are in wav file format.
- Sampling rate of audio samples is 16kHz.

4.3 Impact Analysis

Speech recognition is a technology that helps people to communicate with devices by speaking to them. Speech recognition was created for the sole purpose of generating text from speech, so instead of typing on a keypad, users speak to the computer, which then types the text for them.

4.3.1 Social Impact

Institutions can use speech recognition to authenticate the identities of their clients in order to avoid providing sensitive and risky personal information. The use of voice biometrics in many organisations, such as banks, has helped to reduce fraud and phone crimes.

Speech is extremely useful when our eyes and hands are occupied, such as when driving. Speech recognition technology is now only limited to basic commands for the time being. Researchers will be able to develop more intelligent systems that understand conversational speech as technology progresses. One day, we'll be able to communicate with our machine in the same way as we can with any person.

The Bangla speech to text system will allow Bengalis to communicate with their devices in Bangla. It's pretty easy and thrilling to communicate with a machine in your native language. The speech recognition technology for various languages is improving day by day. As a result, the Bangla STT system should not be left behind.

4.3.2 Ethical Impact

In human-machine speech communication, neither eyes nor arms are needed. Both humans and machines simply talk and/or listen, or the computer performs the task in place of humans. As a result, speech communication may assist individuals who are unable to communicate with their eyes or hands. Both the visually impaired and the physically disabled, as well as the elderly, are included.

The majority of people in Bangladeshi villages do not know how to use technologies. They are also illiterate. As a result, a Bangla speech to text conversion system would allow them to communicate with devices in their own language, even if they have no prior experience with technology.

4.4 Experimental set up

It's vital to have a good experimental setup in order to incorporate the architecture. Continuous power supply is needed for training 215+ hour audio data. To train a deepspeech2 model, a computer with Nvidia GPU support is needed. Ubuntu is the preferred operating system. It is necessary to install the correct CUDA version. All of the work is completed within one environment. So, before running any command, double-check to see if the correct environment is being activated. The GPU driver version should be up to date. The NVIDIA System Management Interface (nvidia-smi) is a command-line tool for controlling and tracking NVIDIA GPUs that runs on top of the NVIDIA Management Library (NVML). This command must be reviewed in the terminal before moving on to the main implementation. The nvidia gpu can work properly with the executing code if this command returns no errors. Since deepspeech2 architecture requires a large amount of memory, enough space in the working directory is required.

4.5 Evaluation of Framework

For any neural network-based architecture, we use training data to train the model and testing data to validate it. Speech from both training and test data was used to evaluate the model. The model was also tested against audio samples that were manually collected and were not part of the data set.

4.5.1 Evaluation with Training Data and Testing Data

The data set was divided into three sections for training, validation, and testing (such as in Table 4.5). Three CSV files containing the file paths for the training, validation, and testing data were developed. The Deepspeech2 architecture accepts two CSV files containing the training and validation data file paths. Training and validation data are transmitted across the network though the file paths of the CSV files. Per row in the CSV files has an audio data path and a corresponding text transcript data path. As a result, an audio sample with its corresponding text transcript is passed through the network for training, row by row tracking the CSV file.

Table 4.1: Splitting Data set

Data Category	Percentage of Splitting	Amount of Audio samples and text transcript
Training Data	75%	164028
Validation Data	12.5%	27337
Testing Data	12.5%	27338

When the number of epochs grows, the accuracy of the train and validation also increases (such as in Fig.4.1). The model performs well when evaluated with the training audio samples of the same data set.

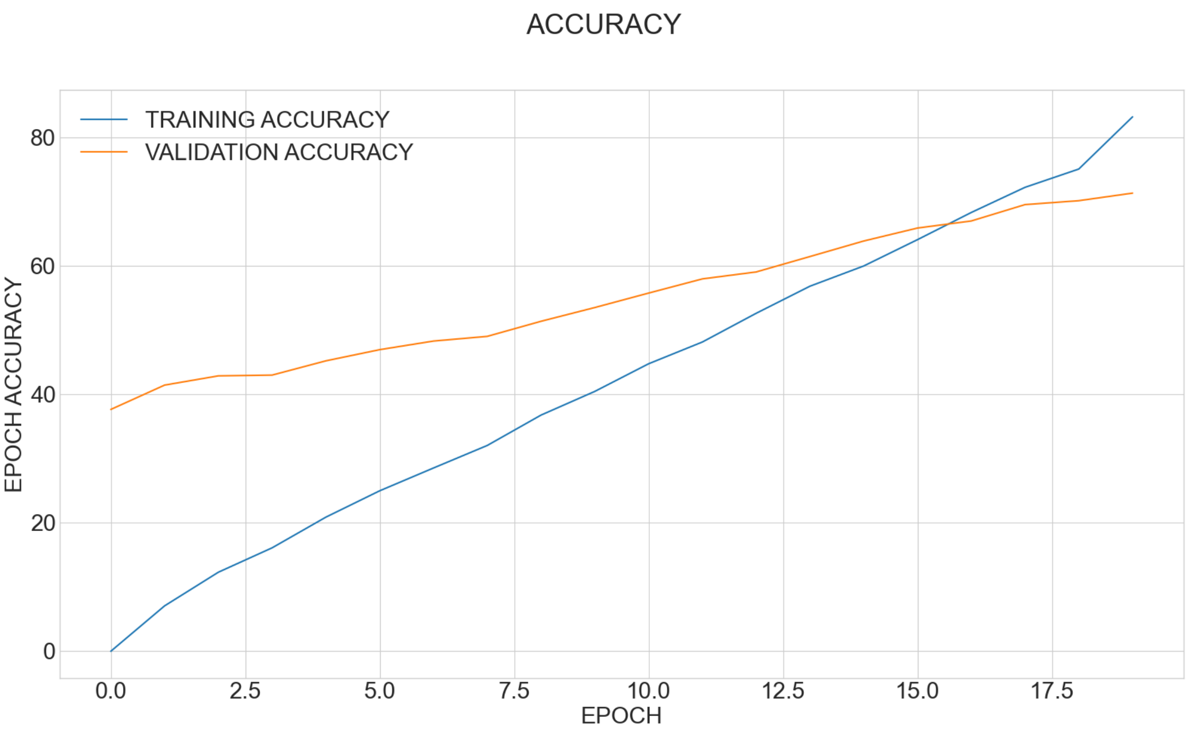
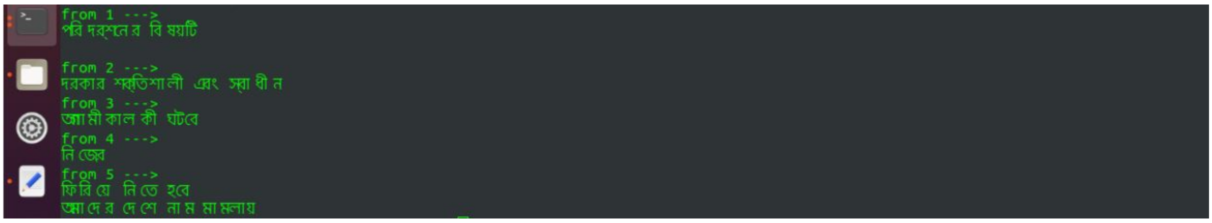


Figure 4.1: Training and Validation Accuracy with respect to Number of Epochs

Many audio samples from train and test data set were used for evaluating the model. Audio samples of following sentences were also passed.

- পরিদর্শনের বিষয়টি
- দরকার শক্তিশালী এবং স্বাধীন
- আগামীকাল কি ঘটবে
- নিজের কৃষ্ণচূড়া
- ধুয়ে নিতে হবে
- আমাদের দেশের নাম বাংলাদেশ

The output text transcripts for these audio samples are shown in Fig.4.2



```
from 1 --->
পরি দর্শনের বি ষয়টি

from 2 --->
দরকার শক্তিশালী এবং স্বা ধী ন
from 3 --->
আগামীকাল কী ঘটবে
from 4 --->
নি জের
from 5 --->
ফিরি য়ে নি তে হবে
আমা দে র দে শে না ম মা মলায়
```

Figure 4.2: Model Output for Audio Samples of Data set

4.5.2 Observation

- First 3 text transcripts are exactly like the audio samples was passed.
- Model could write Bangla conjunctions.
- In 4th sample one word is completely missing in the text transcript. The audio of last word of the sentence is not traced by the model at all.
- In last 2 samples we can see that one word of both sentences is altered by completely different words.

4.6 Evaluation of Performance

The model's output is assessed by comparing it to a recorded audio sample that is entirely unrelated to the data set used. Analyzing the model's performance for female and male audio samples. A comparison is made between the outputs of Google API Bangla STT and the results of our model. Word Error Rate is used to compare the difference between the real output and the predicted output.

4.6.1 Evaluation with Random Audio Samples

We tested the model's output with some randomly recorded audio samples to see how well it worked. The recordings were not made in a noise-free setting. No professional speaker or high-quality microphone was used to capture the audio samples. As a result, this assessment was conducted using real-world audio samples.

Audio recordings of multiple individuals were used to train this model. As a result, the model's prediction must be independent of the speaker. Both female and male audio samples were included in the data collection. As a result, the model's result should be gender-neutral. A man and a female said the same sentences. As a result of this evaluation, we were able to see if the model produces results for any speaker of any gender.

4.6.1.1 Checking the Audio Sample before Prediction

The model was trained on audio files in the.wav format with a sampling rate of 16kHz. As shown in Fig.4.3, before testing the model with any audio sample, the format and sampling rate of the audio sample must be tested to see whether they are suitable. If not, convert the audio to.wav format and adjust the audio sample's sampling rate.If all checks out, the audio sample is passed through the trained model, and the model predicts text output.

We used Librosa, a python package for music and audio processing, to change the sampling rate of the audio sample.

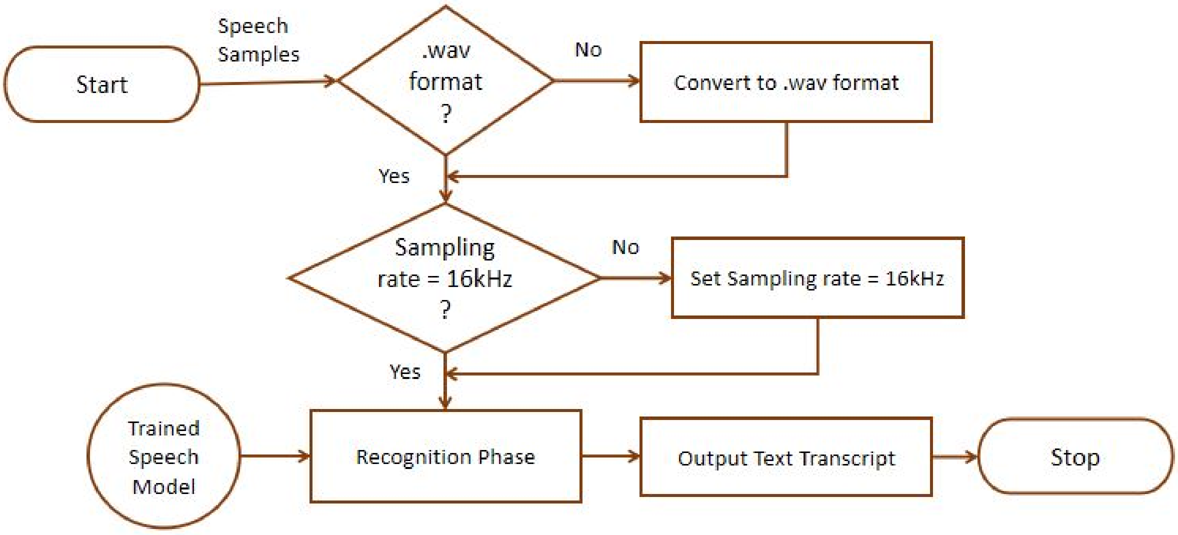


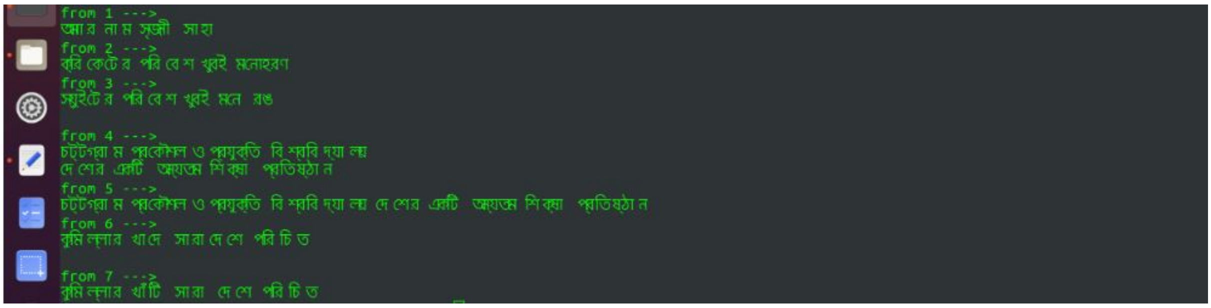
Figure 4.3: Model Evaluation with Random Audio Sample

4.6.1.2 Output of the Model

Audio samples of following sentences were passed.

- আমার নাম সৃজনী সাহা
- চুয়েটের পরিবেশ খুবই মনোরম
- চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয় দেশের একটি অন্যতম শিক্ষাপ্রতিষ্ঠান
- কুমিল্লার খাদি সারা দেশে পরিচিত

The final three audio samples were captured twice, once by a woman and once by a man. The output text transcripts for these audio samples are shown in Fig.4.5



```
from 1 --->
আমার নাম সৃজনী সাহা
from 2 --->
ক্রিকেটের পরিবেশ খুবই মনোহরণ
from 3 --->
স্যুইটের পরিবেশ খুবই মনে রঙ

from 4 --->
চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয়
দেশের একটি অন্যতম শিক্ষা প্রতিষ্ঠান
from 5 --->
চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয় দেশের একটি অন্যতম শিক্ষা প্রতিষ্ঠান
from 6 --->
কুমিল্লার খাদে সারাদেশে পরিচিত

from 7 --->
কুমিল্লার খাঁটি সারা দেশে পরিচিত
```

Figure 4.4: Model Output for Random Audio Samples

The output texts seem to be corrupted due to Ubuntu terminal. The output text transcripts by model are given below to help you understand the model's output clearly.

- Female Speaker : আমার নাম সৃজনী সাহা
- Male Speaker : ক্রিকেটের পরিবেশ খুবই মনোহরণ
- Female Speaker : স্যুইটের পরিবেশ খুবই মনে রঙ
- Male Speaker : চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয় দেশের একটি অন্যতম শিক্ষা প্রতিষ্ঠান
- Female Speaker : চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয় দেশের একটি অন্যতম শিক্ষা প্রতিষ্ঠান
- Male Speaker : কুমিল্লার খাদে সারাদেশে পরিচিত
- Female Speaker : কুমিল্লার খাঁটি সারা দেশে পরিচিত

4.6.1.3 Observation

- The model produced perfectly accurate results for the first audio sample.
- For the third sentence, two audio samples of female and male were fed into the model. Male and female voices have different frequencies. The model produced 100% accurate results for both male and female audio samples. As a result, the model is fully gender neutral.
- Model could write Bangla conjunctions properly even for random audio sample.

- For both male and female, the performance for the second and fourth sentences was inaccurate. Only one character is altered in the fourth sentence production. Two words could not be recognized correctly in the second sentence's prediction.
- For different speakers of different genders, the model could predict text output. As a result, the model is speaker and gender independent.

4.6.2 Word Error Rate

At the word stage, the Speech to text conversion system can make three types of errors: deletion, insertion, and substitution. The Word Error Rate (WER) is used to assess the ASR system's accuracy. The lower the WER, the more efficient the machine is. WER is calculated by dividing the sum of the all 3 types of errors by the total number of words in the reference word transcription, expressed in percent.

The equation of WER is given below.

$$WER = (I + D + S)/ N$$

Here

I = No of Insertions

D = No of Deletions

S = No of Substitutions

N = No of words in the Original Text

Substitution : When a term is changed, it is called a substitution.

Insertion : When a word is inserted that wasn't said in the audio sample, it's called an insertion.

Deletion :When a word is entirely removed from a transcript, it is called a deletion.

Observing the errors of the previously mentioned model output's, substitution , insertion and deletion is presented in Table 4.2

Table 4.2: Error observation

Original Sentence	Model Predicted Sentence	Error Observation	Error Type
নিজের কৃষ্ণচূড়া	নিজের	Last word is missing	Deletion
কুমিল্লার খাদি সারা দেশে পরিচিত	কুমিল্লার খাঁটি সারা দেশে পরিচিত	One character is altered	Substitution
আমাদের দেশের নাম বাংলাদেশ	আমাদের দেশের নাম মামলায়	Completely new word inserted	Insertion

WER between the reference sentence and the model predicted output sentence can be measured by using JiWER which gives a assessment of automated speech recognition using similarity tests.

WER comparison of audio samples recorded by male and Female speaker is shown in Table 4.3

Table 4.3: WER observation

Original Sentence	Male Speaker's output	WER	Female speaker's output	WER
কুমিল্লার খাদি সারা দেশে পরিচিত	কুমিল্লার খাদে সারা দেশে পরিচিত	20%	কুমিল্লার খাঁটি সারা দেশে পরিচিত	20%
চুয়েটের পরিবেশ খুবই মনোরম	ক্রিকেটের পরিবেশ খুবই মনোহরণ	50%	স্যুইটের পরিবেশ খুবই মনে রঙ	65%

When the number of epochs grows, WER decreases (such as in Fig.4.5).

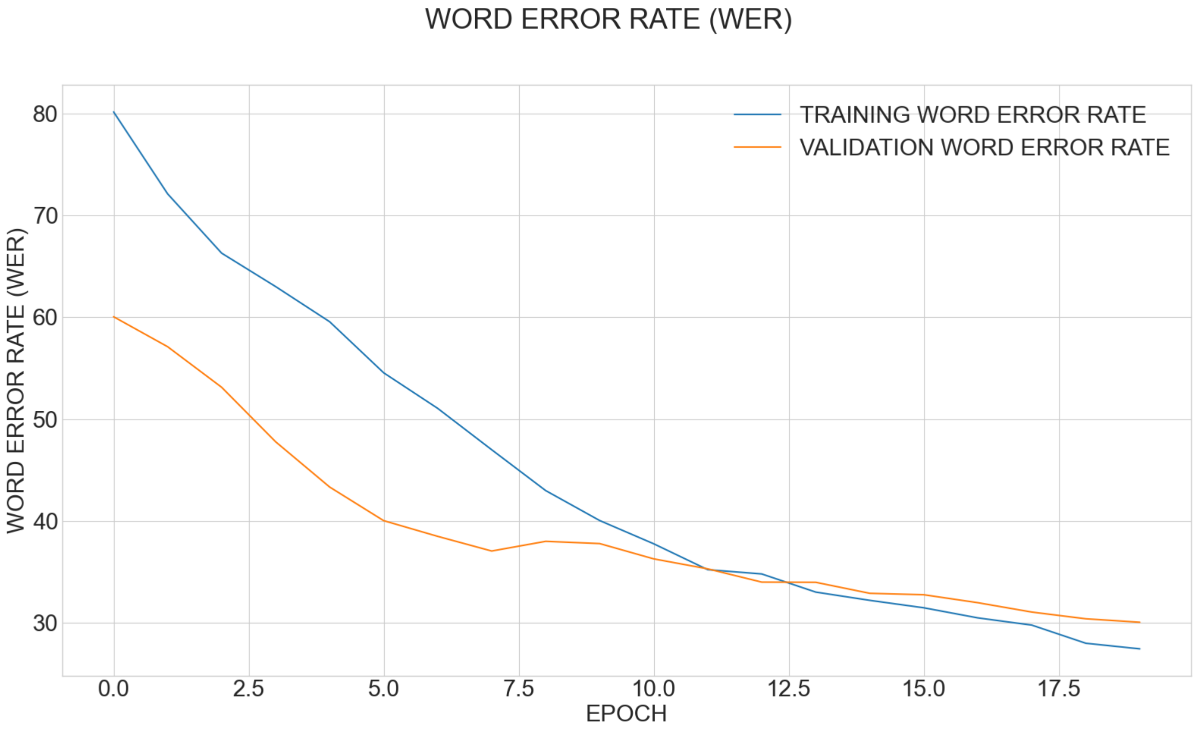


Figure 4.5: WER with respect to Number of Epochs

4.6.2.1 Comparison with Google API Bangla STT

Comparison of the result of the model with Google's speech to text API for bangla with same inputs and observed the text transcripts. Compared both model with respect to WER of the models' outputs.

Audio samples of following sentences were passed through both Google API and implemented architecture.

- Audio Sample 1 : জোড়ালো করতেই ওবামাা
- Audio Sample 2 : যাতে বিবাহ শাদী দিলে
- Audio Sample 3 : বিবাহ ব্যাপারটাকে নিয়ন্ত্রিত করায়
- Audio Sample 4 : প্রেসিডিয়ামে ঠাঁই দিয়েছিলেন
- Audio Sample 5 : অতঃপর ১৯৪৭ সালের ২ জুলাই

Comparative study of output of both architectures is shown in Table 4.4

Table 4.4: Comparison between Implemented architecture and Google API Bangla STT

Audio Sample	Google API's Output	WER	Implemented Architecture's Output	WER
01	জোরালো করতে ওবামা	66.6%	জোড়ালো করতে ওবামা	33%
02	যাদের বিবাহ শাদী দিলে	25%	বিবাহ শাদী দিলে	25%
03	বিবাহ ব্যাপারটাকে নিয়ন্ত্রিত করা	25%	বিবাহ ব্যাপারটা নিয়ন্ত্রিত	50%
04	প্রেসিডিয়ামের ঠাই দিয়েছিলেন	67%	প্রেসিডিয়ামে ঠাই দিয়াছিলেন	67%

Interesting thing about the last audio sample is when we utter a digit it can be written in both digits or characters.

Output of Google API : অতঃপর হাজার 947 সালের 2 জুলাই

Output of Implemented Architecture : অতঃপর উনিশশো সাতচল্লিশ সালের দুই জুলাই

Here we can see that Google API outputs digits in English. Our implemented architecture gives output in characters.

4.7 Discussion

4.7.1 Hardware limitation

The architecture used in this project is computationally expensive. Our laptop which has only 4 GB RAM was unable run such an architecture. We tried to work in a PC in our department which has 2 GB GPU support. But the PC was not working well. We faced problems during installing the dependencies properly. Some how we have managed to complete our training in a 8 GB RAM PC integrated with GPU Nvidia GTX 1080 GeoForce (8GB) . Only 8GB GPU memory was still very less than required and We could only train the model with batch size 8 whereas at least a batch size of 32 is recommended. It took so many days to train the model with 218k data. The preparation of the audio files for training was done in our laptop which was so time consuming. Just for converting 218k audios' file format to .wav format, it took 12 days. A detailed hardware requirement is given in the following Table 4.5

Table 4.5: Detailed Hardware Requirement

	Minimum	Recommended
Processor Type	Core-i5	Core-i7
Clock Speed	2.8 GHz	3.1 GHz
RAM	8GB	32GB or more
Hard Drive Space	40 GB	40 GB or more
OS	64 Bit Ubuntu	64 bit Ubuntu
USB Port	2.0	3.0
GPU	NVIDIA chip set with memory 8GB	NVIDIA chip set with memory 16GB or more

Due to hardware constraints and the fact that we could only work in the PC during the office time only achieving the desired result became more challenging.

4.7.2 Studying the training data

There are four major factors for the speech database

- One is the size of the speech corpus. As deep learning model just learns from data, so amount of data is very important issue. Recognizing speech is very tough because speech can be taken from people of different ages or different genders. So huge amount of data with a huge variety is essential for speech to text conversion system. It is recommended to use a speech corpus having 7000+

hour data to train the model [9]. For recognizing English the DS2 architecture was trained on 11,940 hours of speech [3]. But for Bangla the only open sourced speech corpus we managed to find that is Bengali ASR corpus with 215+ hour speech data only. So the amount of speech data used in this project was really too small.

- Second factor is type of speech data. For building a noise robust speech to text conversion system the model has to be tested on noisy speech. Data set should include various noisy background, like the bus, cafe, street and pedestrian areas. Speech recognition for such setting is a significant task, networks need the capacity to manage a range of situations in real-world settings for greater generalization [18]. The data set we used contains a little background noise which is not adequate to train the model to make a robust speech recognition system.
- Third one is the length of the audio sample. A successful speech recognition system includes speech sample of different duration. Speech recordings should be ranged from a few minutes to more than a hour [3]. But Bengali ASR data set includes samples of few minutes or few seconds.
- Fourth aspect is various linguistic accents. The audio samples should be taken from people of different area. Since some people like in Sylhet or Chittagong speak Bengali with somewhat different accent. So making a perfect speech recognizer the data set should contain accented speech for training the model. Bengali ASR data set does not contain accented speech.

4.8 Conclusion

The performance of the Bangla speech to text conversion system is shown in this chapter. The suggested methodology's performance is also discussed. The thesis work is brought to a close in the next chapter.

Chapter 5

Conclusion

5.1 Conclusion

In today's technology, speech to text conversion system has become an essential part. Speech to text for Bangla is an ongoing research field. Using statistical parametric approach speech to text conversion system for Bangla has been researched and developed. Deep learning is a buzzword nowadays and they help build complex architecture like STT with much simplification even though they are computationally heavy. There is a few work deep learning based STT system for Bangla Language. The implementation of speech-to-text for Bangla is a step towards that. Here, we implemented the DeepSpeech2 architecture to fit for the Bangla language and trained the model with Bengali ASR speech corpus and corresponding text transcripts correcting the error in the data set. Then the result from the model is evaluated and word error rate for different sample input was observed. We believe that this strategy of Bangla speech to text conversion system will continue to evolve by rising computing capacity and data set sizes.

5.2 Future Work

The following enhancements can be further looked into:

- Training the DeepSpeech2 architecture for larger database than Bengali ASR database and examining result.
- Implementing DeepSpeech2 architecture for database of multiple speaker of different ages and genders.
- Training the model with noisy Bengali speech samples for making noise robust speech recognition system.

- Making data set containing various linguistic accents and audio samples having duration more than a hour with proper text transcripts. Then a better Bengali speech to text conversion system can be made using this data set.

References

[1] *Bangla ranked at 7th among 100 most spoken languages worldwide*, Feb. 2020. [Online]. Available: `https://www.dhakatribune.com/world/2020/02/17/bengali-ranked-at-7th-among-100-most-spoken-languages-worldwide` (cit. on p. 2).

[2] M. M. H. Nahid, M. A. Islam and M. S. Islam, 'A noble approach for recognizing bangla real number automatically using cmu sphinx4,' in *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*, IEEE, 2016, pp. 844–849 (cit. on p. 2).

[3] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, 'Deep speech 2: End-to-end speech recognition in english and mandarin,' in *International conference on machine learning*, PMLR, 2016, pp. 173–182 (cit. on pp. 3, 5, 9, 14, 16–18, 20, 38).

[4] M. M. H. Nahid, B. Purkaystha and M. S. Islam, 'Bengali speech recognition: A double layered lstm-rnn approach,' in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, IEEE, 2017, pp. 1–6 (cit. on pp. 4, 14).

[5] A. K. Paul, D. Das and M. M. Kamal, 'Bangla speech recognition system using lpc and ann,' in *2009 Seventh International Conference on Advances in pattern recognition*, IEEE, 2009, pp. 171–174 (cit. on p. 4).

[6] P. Khilari and V. Bhope, 'A review on speech to text conversion methods,' *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 4, no. 7, pp. 3067–3072, 2015 (cit. on pp. 7, 8, 13).

[7] B. H. Juang and T. Chen, 'The past, present, and future of speech processing,' *IEEE signal processing magazine*, vol. 15, no. 3, pp. 24–48, 1998 (cit. on p. 9).

[8] N. Morgan and H. Bourlard, 'An introduction to hybrid hmm/connectionist continuous speech recognition,' *IEEE Signal Processing Magazine*, vol. 12, no. 3, pp. 25–42, 1995 (cit. on pp. 10, 11).

[9] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, 'Deep speech: Scaling up end-to-end speech recognition,' *arXiv preprint arXiv:1412.5567*, 2014 (cit. on pp. 10, 14, 18, 38).

[10] M. Hasnat, J. Mowla, M. Khan *et al.*, 'Isolated and continuous bangla speech recognition: Implementation, performance and application perspective,' 2007 (cit. on p. 12).

[11] G. Muhammad, Y. A. Alotaibi and M. N. Huda, ‘Automatic speech recognition for bangla digits,’ in *2009 12th International Conference on Computers and Information Technology*, IEEE, 2009, pp. 379–383 (cit. on p. 12).

[12] A. Firoze, M. S. Arifin, R. Quadir and R. M. Rahman, ‘Bangla isolated word speech recognition.,’ in *ICEIS (2)*, 2011, pp. 73–82 (cit. on p. 12).

[13] M. Hossain, M. Rahman, U. K. Prodhan, M. Khan *et al.*, ‘Implementation of back-propagation neural network for isolated bangla speech recognition,’ *arXiv preprint arXiv:1308.3785*, 2013 (cit. on p. 12).

[14] S. K. Roy, A. K. Ghosh, A. S. Asa, M. P. Uddin, M. R. Islam and M. I. Afjal, ‘Bengali consonants-voice to text conversion using machine learning tool,’ *International Journal of Research in Computer Engineering and Electronics*, vol. 6, no. 1, 2017 (cit. on p. 12).

[15] S. Sultana, M. Akhand, P. K. Das and M. H. Rahman, ‘Bangla speech-to-text conversion using sapi,’ in *2012 International Conference on Computer and Communication Engineering (ICCCE)*, IEEE, 2012, pp. 385–390 (cit. on p. 13).

[16] P. Barua, K. Ahmad, A. A. S. Khan and M. Sanaullah, ‘Neural network based recognition of speech using mfcc features,’ in *2014 international conference on informatics, electronics & vision (ICIEV)*, IEEE, 2014, pp. 1–6 (cit. on p. 13).

[17] M. T. Tausif, S. Chowdhury, M. S. Hawlader, M. Hasanuzzaman and H. Heickal, ‘Deep learning based bangla speech-to-text conversion,’ in *2018 5th International Conference on Computational Science/Intelligence and Applied Informatics (CSII)*, IEEE, 2018, pp. 49–54 (cit. on p. 14).

[18] S. H. Sumit, T. Al Muntasir, M. A. Zaman, R. N. Nandi and T. Sourov, ‘Noise robust end-to-end speech recognition for bangla language,’ in *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, IEEE, 2018, pp. 1–5 (cit. on pp. 14, 17–19, 24, 38).

[19] A. Graves, ‘Connectionist temporal classification,’ in *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, 2012, pp. 61–93 (cit. on p. 20).

[20] *Crowdsourced bengali bangladesh [bn-bd] asr dataset.* [Online]. Available: `https://research.google/tools/datasets/bengali-asr/` (cit. on pp. 24, 26).