

# **APPLICATIONS OF END TO END AUTOMATIC SPEECH RECOGNITION**

**A Thesis Submitted  
In Partial Fulfillment of the Requirements  
for the Degree of**

**MASTER OF TECHNOLOGY**  
**in**  
**VLSI DESIGN AND EMBEDDED SYSTEM**  
**by**

**RUPESH KUMAR**  
(Roll No. 2K22/VLS/12)

**Under the Supervision of**

**Mr. Piyush Tewari**  
(Assistant Professor, Department Of ECE)

**Mr. Sumit Khandelwal**  
(Assistant Professor, Department Of ECE)



**Department of Electronics and Communication Engineering**

**DELHI TECHNOLOGICAL UNIVERSITY**

**(Formerly Delhi College of Engineering)**

**Shahbad Daultpur, Main Bawana Road, Delhi-110042. India**

**May, 2024**



# **DELHI TECHNOLOGICAL UNIVERSITY**

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

## **CANDIDATE'S DECLARATION**

I Rupesh Kumar, Roll No – 2K22/VLS/12 student of M.Tech (VLSI DESIGN AND EMBEDDED SYSTEM) hereby certify that the work which is being presented in the thesis entitled **“APPLICATIONS OF END TO END AUTOMATIC SPEECH RECOGNITION”** in partial fulfilment of the requirements for the award of the Degree of Master of Technology, submitted in the Department of Electronics and Communication Engineering, Delhi Technological University is an authentic record of my work carried out during the period from 2022 to 2024 under the supervision of Mr. Piyush Tewari and Mr. Sumit Khandelwal.

.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

**Candidate's Signature**



# DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

## **CERTIFICATE BY THE SUPERVISORS**

I certify that RUPESH KUMAR (2K22/VLS/12) has carried out the search work presented in this thesis “**APPLICATIONS OF END-TO-END AUTOMATIC SPEECH RECOGNITION**” entitled for the award of **Master of Technology** from the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi, under my supervision. To the best of our knowledge, the thesis embodies the results of the student's original work and does not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Mr. Piyush Tewari  
Assistant Professor  
Department of ECE  
Delhi Technological University

Mr. Sumit Khandelwal  
Assistant Professor  
Department of ECE  
Delhi Technological University

## ABSTRACT

This project comprehensively investigates the applications of end-to-end ASR, including models like Transformers and the combination of RNNs with CNNs and CTC loss for the English language. The primary goal is to evaluate the performances of these architectures for sequence-to-sequence tasks that require accurate temporal alignment and robust handling of input sequences with varying lengths, specifically in the context of speech recognition. We tried to compare applications of E2E ASR by using RNN-CNN models and transformers models. We used the datasets from LJspeech for the English language. The RNN-CNN model combines the advantages of CNNs for extracting features and RNNs for processing sequential input to enable alignment-free training. The CNN component enhances the encoding of local features, while the RNN component captures temporal dependencies. The combined effect of both components leads to an improvement in recognition accuracy. The second model utilizes a Transformer architecture, which utilises self-attention for capturing long-range dependency without recurrent connections. This architectural design tackles the constraints of RNNs in managing lengthy sequences and parallel processing, resulting in the potential for quicker training and inference durations. The results of our experiments on a commonly used English language dataset namely LJspeech indicate significant performance improvements.

The Transformer model also demonstrates higher scalability and efficiency when dealing with huge datasets. We compared the WER and computation time for both models and found superior WER performance by 3% to 4% for the transformer-based model over the RNN-CNN model. Additionally, the transformer based model was found to be five times more time efficient per epoch but requires more number of epochs for training. The results indicate that RNN-CNN models are efficient for tasks with prominent local dependencies, whereas Transformers exhibit notable benefits in terms of computational efficiency and managing long-range dependencies. This makes Transformers a compelling option for large-scale English language processing applications.

## **ACKNOWLEDGEMENT**

I would like to sincerely thank all persons who have made significant contributions to the effective completion of this thesis. Initially, I want to convey my deep appreciation to my advisers, Mr. Piyush Tewari and Sumit Khandelwal, for their consistent support, priceless assistance, and perceptive criticism during this research endeavour. Their extensive expertise and unwavering assistance have been vital in shaping this piece of work. I would like to sincerely thank Mr. Piyush Tewari and Mr. Sumit Khandelwal, for my thesis work, for their astute comments and invaluable analysis. Their contributions have significantly improved the overall quality of this thesis. I really appreciate my friends for their companionship and for fostering a fascinating and supportive atmosphere.

Rupesh Kumar

(2K22/VLS/12)

VLSI DESIGN AND EMBEDDED SYSTEM

## TABLE OF CONTENTS

PARTICULARS	PAGE NO.
<b>Candidate's Declaration</b>	<b>ii</b>
<b>Certificate</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Acknowledgement</b>	<b>v</b>
<b>Table Of Contents</b>	<b>vi</b>
<b>List Of Figures</b>	<b>vii</b>
<b>List of Abbreviations</b>	<b>viii</b>
<b>CHAPTER 1: INTRODUCTION</b>	<b>1</b>
<b>CHAPTER 2: LITERATURE REVIEW</b>	<b>3</b>
2.1 Deep learning and end-to-end ASR	3
2.2 Acoustic model $P(X S)$	4
2.3 End to End Model	5
2.4 Research Gaps	6
<b>CHAPTER 3: METHODS AND TECHNIQUE</b>	<b>8</b>
3.1 CNN-RNN MODEL WITH CTC LOSS	8
3.2 TRANSFORMER WITH CTC LOSS	10
3.2.1 1Encoder-Decoder with Attentions	12
3.2.2 Multi-Headed Attention	12
3.2.3 Layer Architecture	12
3.3 ASR training and decoding	12
<b>CHAPTER 4: EXPERIMENTAL RESULTS</b>	<b>14</b>
4.1 DATASET	14
4.2 TRAINING AND RESULTS	17
4.3 Result for RNN-CNN MODEL	17
4.4 Result for Transformer Model	18
<b>CHAPTER 5: CONCLUSION</b>	<b>20</b>
<b>References</b>	<b>21</b>

## LIST OF FIGURES

Figure 2.1: Functional Structure of End-to-End Model .....	5
Figure 3.1: CNN-RNN Network sample for end-to-end ASR layers .....	7
Figure 3.2: Deepspeech2 .....	8
Figure 3.3: Proposed CNN Model.....	9
Figure 3.4: Architecture Model of Transformer.....	19
Figure 4.1: Speech To Text Transcription .....	13
Figure 4.2: Audio spectrogram.....	15
Figure 4.3: Deepspeech 2 Model .....	16
Figure 4.4: Loss vs epoch.....	17
Figure 4.5: wer vs epoch .....	18
Figure 4.6: Epoch vs loss.....	19

## LIST OF ABBREVIATIONS

STT:	Speech To Text
ASR:	Automatic Speech Recognition
CNN:	Convolutional Neural Networks
CTC:	Connection Temporal Classification
HMM:	Hidden Markov model
RNN:	Recurrent Neural Network
LSTM:	Long Short-Term Memory
PDLSTM:	Parallel Time-Delayed
WER:	Word Error Rate
S2S:	Sequence to Sequence
GMM:	Gaussian Mixture Model
NMT:	Neural Machine Translation
LVSCR:	Large-vocabulary continuous speech recognition
E2E:	End to End



# CHAPTER 1

## INTRODUCTION

ASR is a technological advancement which allow machines in comprehend and analyze human speeches. It entails the transformation of oral communication into written form, enabling smooth interaction between humans and computers. ASR systems play a fundamental roles in a wide range of applications, including technologies that are voice-activated, such as Alexa and Siri, as well as services that transcribe audio, and software that automates customer support processes. Recently, there is notable developments in the progress of ASR systems, which have made them more reliable and broadened their practical uses. The advancement can be mostly attributed to the growing utilization of speech as an intuitive interface between humans and computers, as well as the accessibility of extensive datasets. Recent advancements in training approaches, like Seq2Seq and CTC, has facilitated the advancement in state of the art ASR transport network. CNN and RNN are the deep learnings technique that are utilized by these transport networks. This study examines efficacy and feasibility of modern deep learning-based, end-to-end ASR models. More precisely, it investigates the utilization of CNN-RNN hybrid systems that is trained end-to-end[1] having CTC Loss in the English ASR task.

HMM-depended ASR systems have consistently delivered top-notch performance for many decades. [2] [3]. In recent times, E2E ASR models have shown impressive results by tackling the task of converting speech into text using a single S2S system [4]. Attention based encoder decoder architectures, RNN-T), and CTC are the most efficient and often employed technique of E2E ASR [5],[6]. Online and streaming applications achieve high performance in ASR with the use of RNN-T based ASR systems. These systems have been successfully implemented in production systems [6], [7].

The neural network architecture is crucial for achieving high-quality ASR performance, and it is equally vital as the end-to-end ASR modeling technique- RNN architectures LSTM neural network is commonly employed in E2E ASR models. Out of the RNN-based systems, BLSTMs achieve the best results. However, they are not suitable for streaming applications. Instead, one-directional LSTMs or LCBLSTMs should be employed, as suggested by [8]. Compared to the LC-BLSTM, the PTDLSTM architecture features a lower WER gap between unidirectional and bidirectional systems and a higher computational complexity. Moreover, the Transformer model, a self-attention based encoder-decoder architecture initially designed for machine translation, has been applied to ASR recently. In comparison to designs based on RNNs, this application has produced WERs that are superior[9].This study focuses on enhancing a streaming Automatic Speech Recognition (ASR) application by including the Transformer concept into both an encoder and decoder type attention systems in model. In addition, the encoder utilizes time restricted self attention. In order to properly integrate the Transformer-Attention (TA) approach and achieve optimal outcomes in training and decoding, the model is trained concurrently with a CTC goal [10].Employing the frame-synchronously one pass decode method, which involves the simultaneous decoding and scoring of CTC-transformer.

The significance of depth in achieving efficient end-to-end ASR models utilizing the Transformer is paramount .Additionally, we conduct an extensive comparision studies between Transformer and the RNN, revealing notable improvements in performances, especially in tasks belongings to Automatic Speech Recognition (ASR).

This study offers a comprehensive examination of the entire E2E model. We demonstrate the progression of E2E technology, assess the advantages and disadvantages of several end-to-end

technology frameworks, and offer a concise comparison of RNN based and Transformer-based E2E models. I have employed the WER as a means of evaluating the caliber of the model. The WER is determined by aggregating the amount of insertions, deletions, and substitutions that occur inside a recognized sequence of words, and then dividing that sum by the total number of words initially pronounced. The WER will assist us in assessing the quality of the model. Finally, I offer my insights and conclusions for future projects.

## **Overview**

This report is structured into 4 further chapters ,whose objectives are

1. Chapter 2: Discuss the key insights obtained from literature survey
2. Chapter 3: Defines the proposed Network model and formulates the problem statements and  
Analytical solution
3. Chapter 4: Discusses about the results that is obtained.
4. Chapter 5: Concluding the report and laying the scope of future work.

## CHAPTER 2 LITERATURE REVIEW

This chapter explores the research papers, articles etc. that were examined to acquire insights into current advancements in E2E ASR and the formulation of the problem statement. The papers serve the purpose of conforming to industry and academia requirements and comprehending the current research deficiencies.

### Literature Survey

#### 2.1 End to end ASR and Deep Learning:

HMM-GMM acoustic models, which employed HMMs for accurate alignment and GMMs for associating compatible tri-phones with written characters, plays a pivotal role in the advancement of conventional ASRs. These systems held a dominant position in the market for a period of time before being surpassed by the HMM-DNN system, which utilizes deep neural networks instead of GMMs. The combination of (Seq2Seq) systems and CTC models has significantly transformed the training approach for speech recognition systems. This approach involves to train a single network which maps directly to audio sequences to the corresponding text output. The E2E models, utilizing RNN [11], [12], attention model[13], [14], and other models, attained exceptional performance on various dataset benchmarks. Mostly contemporary deep learning based E2E systems input log-spectrogram or MFCC properties into their networks. In recent times, there has been a shift towards focusing on extracting features directly from the raw audio output. This is because deep neural networks possess an innate capability to learn and understand the properties present in raw inputs. Recently, a proposed approach to do this involves using trainable filter banks that are based on time-delay convolution. These filter banks have been found to perform similarly to Mel filter bank-based systems on the WSJ dataset. The HMM-based model has traditionally the most efficient LVCSR model for achieving accurate recognition results. The HMM-based model is divided in three distinct components: an acoustic system, pronunciation system and a language system. Every each component serves a unique purpose and is independent from the rest.

An acoustic model is tasked with transforming speech input in a sequence of features, typically phoneme and sub phoneme. The pronunciation model, typically created by proficient linguists, seeks to establish a connection between these phoneme and sub phoneme and grapheme. The language modeling directs these character sequences to a fluid final transcriptions [15]. In this HMM, sound is the observable variable and its hidden state represents the underlying characteristic. HMM-based models are used to represent an HMM with a state set  $\{1, \dots, J\}$ . In this HMM, sound is observed and its hidden state represents its characteristic. Models in the HMM with a set  $\{1, \dots, M\}$  based on HMMs decomposes  $p\left(\frac{K}{X}\right)$ .

$$\arg \max_K p\left(\frac{K}{X}\right) = \arg \max_{K \in Z^+} \frac{p(K, X)}{p(X)} \quad (1)$$

$$= \arg \max_{K \in Z^+} p(K, X) \quad (2)$$

$$= \arg \max_{K \in Z^+} \sum_s p(K, X, S) \quad (3)$$

$$= \arg \max_{K \in Z^+} \sum_s p\left(\frac{X}{S}, K\right) p(S, K) \quad (4)$$

$$= \arg \max_{K \in Z^+} \sum_s p\left(\frac{X}{S}, K\right) p\left(\frac{S}{K}\right) p(K) \quad (5)$$

Conditionally independent hypothesis states that we may approximate to  $p(X|S, K) \approx p(X|S)$ . Consequently,

$$\arg \max_K p\left(\frac{K}{X}\right) \approx \arg \max_{K \in Z^+} \sum_s p\left(\frac{X}{S}, K\right) p\left(\frac{S}{K}\right) p(K) \quad (6)$$

These three factors  $p(X|S)$ ,  $p(S|K)$ , and  $p(K)$  correspond to an acoustics model, pronunciations model and languages model.

## 2.2 Acoustic model $P(X|S)$ :

Given the hidden sequence  $S$ , it measures the likelihood of seeing  $X$ . The probability  $P(X|S)$  can be found in Hidden Markov Models (HMMs) by applying the chain rule of probability and assuming that observations in HMMs depend solely on the hidden state at any one time. Given an event  $S$ , the conditional probability of occurrence  $X$  is  $P(X|S)$ . Is potentially decomposing in the following parts:

$$p\left(\frac{X}{S}\right) = \prod_{i=1}^T p(x_i / x_1, \dots, x_{i-1}, S) \quad (7)$$

$$\approx \prod_{t=1}^T p\left(\frac{x_t}{s_t}\right) \propto \prod_{t=1}^T \frac{p(s_t/x_t)}{p(s_t)} \quad (8)$$

For the acoustic model,  $p\left(\frac{x_t}{s_t}\right)$  is observational probability, which is represented generally by GMM. The PDF of a hidden states  $p(s_t/x_t)$  is computed using the DNN method. This leads to two distinct models for  $P(X|S)$ : HMM-GMM and HMM-DNN. Traditionally, the HMM-GMM method has been the conventional framework for voice recognition. Nevertheless, as deep learning progressed, speech recognition for acoustic modelling started integrating DNNs [12]. This approach uses a DNN to determine the HMM state's posterior probability. It can replace the traditional GMM observation probabilities with this probability by transforming it into likelihood [16]. As a result, the HMM-GMM model has developed in the HMM-DNN model [17], which surpasses the HMM-GMM model and has become the most advanced technique in ASR.

The HMM-based model integrates several technologies and achieves certain goals through its multiple components. Frame-by-frame dynamic time warping is a significant application of HMMs. The emission probabilities of the hidden states in the HMM are calculated using either GMM or DNN methodologies. The design and operational strategies of the HMM based model determine how it addresses challenges in practical applications.

The training process is complex and rigorous in order to optimize on a global scale. The HMM-based model often employs various training procedures and datasets for its different modules. Each module is independently optimized with its own objective functions, which typically do not align with the real

criteria used to assess the effectiveness of LVCSR. Hence, even if every module operates optimally, it does not guarantee overall global optimization. The HMM-type model utilizes the assumptions of conditional independence, both across modules and within a module, to streamline the training process and produce the model. This is incongruent with the factual circumstances of LVCSR.

### 2.3 END TO END MODEL

As previously described, there are limitations in the HMM-based model, as deep learning methods progress, have resulted in a growing number of research papers focusing on end-to-end LVCSR. An E2E model refers to a system which immediately converts an audio input sequence into either a word or another sequence of graphemes.

In Figure 2.1, its functional structure is displayed.

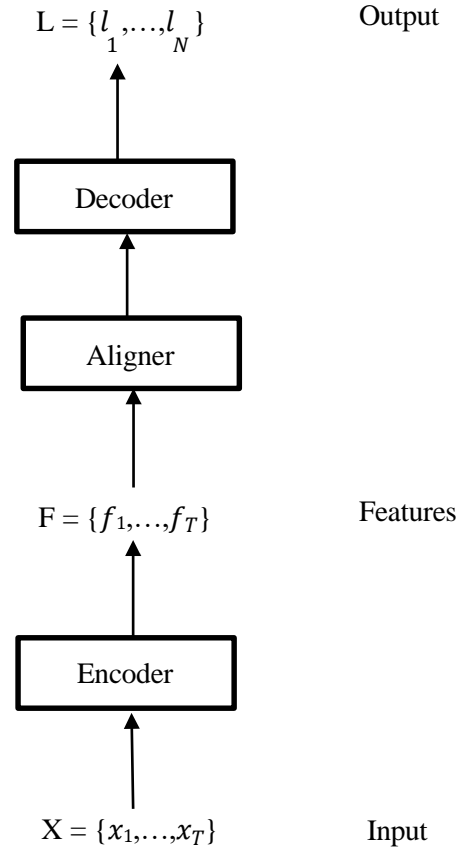


Fig 2.1. Functional Structure of E2E models [4]

The constituents of the vast majority of E2E speech recognition models are as follows: The encoder maps speech input sequences to feature sequences, the aligner aligns language and feature sequences, and the decoder decodes the final identification result. It is important to understand that this division may not always happen because an engineered modular system is a complete structure when seen from start to finish, and it is usually challenging to discern which part conducts each sub-task.

The end-to-end approach differs from the HMM-based model by utilizing a deep network to directly associate auditory inputs with label sequences, eliminating the requirement for meticulously built intermediate states. Furthermore, there is no need for any subsequent processing of the output.

Both Hidden Markov Model (HMM)-based and end-to-end models face challenges when it comes to aligning data, particularly in the context of voice recognition. An issue that often arises is the task of finding the appropriate alignment between a label in the sequence and the corresponding speech data. End-to-end models employ soft alignment, wherein each audio frame is probabilistically linked to all potential states without a mandatory, explicit connection.

The end-to-end model can be classified into one of three categories based on the implementation of soft alignment:

- **CTC-based:** The process starts by enumerating all possible exact alignments (represented as paths), and then CTC combines these alignments to get a flexible alignment. The CTC algorithm calculates the number of hard alignments on the assumption that the output labels are independent of each other.
- **The RNN-transducer algorithm** calculates all possible hard alignments and then combines them to obtain a soft alignment. However, in terms of defining the path, RNN-transducer is different from CTC because it does not make separate assumptions about labels while calculating hard alignments and probability computation.
- **Attention Based:** This approach utilizes the Attention mechanism to compute the soft alignment information between the input data and the resulting label, instead of enumerating all possible hard alignments.

## 2.4 Research Gaps

HMMs and GMMs are widely recognized techniques in the fields of statistical modeling and machine learning. However, despite their extensive use, there are still several unresolved research requirements and opportunities for enhancement. Some of the primary research gaps in these models are adapting HMMs for new domains with limited labeled data and developing transfer learning algorithms specifically designed for HMMs.

### 1. Restricted comprehension of the context.

HMM and GMM models predominantly depend on processing each frame individually, which restricts their capacity to grasp distant relationships in speech. They consider each frame to be conditionally independent of others, disregarding the wider context that could enhance recognition accuracy.

### 2. Assumption of Gaussian Distributions

The GMM component postulates that the data can be represented by a combination of Gaussian distributions. This assumption may be excessively naive, as real-world speech samples frequently display more intricate patterns that Gaussian mixtures are unable to accurately represent.

### 3. Problems in handling sentences of long length

As the speech datasets become more sophisticated and larger in size, HMM-GMM models face challenges in terms of scalability. Training these models on big datasets is computationally demanding and time-consuming, which restricts their practical usability in contemporary voice recognition tasks.

#### 4. Challenges in Optimization

The training method for HMM-GMM models entails separately optimizing numerous modules, each with its own distinct objective function. This modular optimization method does not provide global optimality, resulting in inferior overall performance. The absence of end-to-end optimization can impede the model's capacity to get the highest level of recognition accuracy.

## CHAPTER 3

### METHODS AND TECHNIQUES

#### 3.1 CNN-RNN MODEL WITH CTC LOSS

In the next subsections, we present information about the evaluated EESR networks that utilize a combination of CNN and RNN, together with the use of CTC loss.

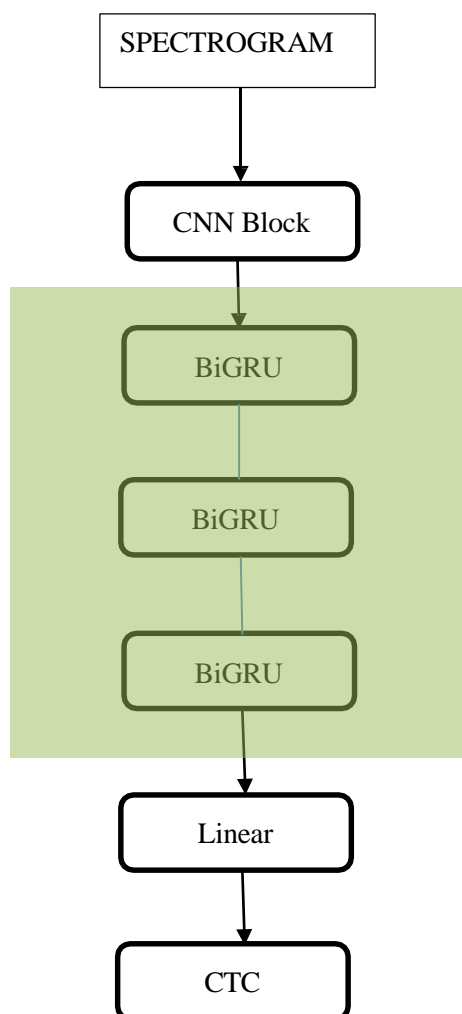


Fig 3.1 Sample CNN-RNN Network for end-to-end ASR with 3 BiGRU layers[1]



Figure 3.1 explains overall structure of the CNN-RNN based EESR models, which have been increasing popularly in recent research[1][12]. The convolution block stands for an n-layer convolution stack [1] Figure 3.2 shows the first 2-layer convolution stack. Hard Tanh non-linearity and Batch Normalization layers follow each convolutional layer [13].

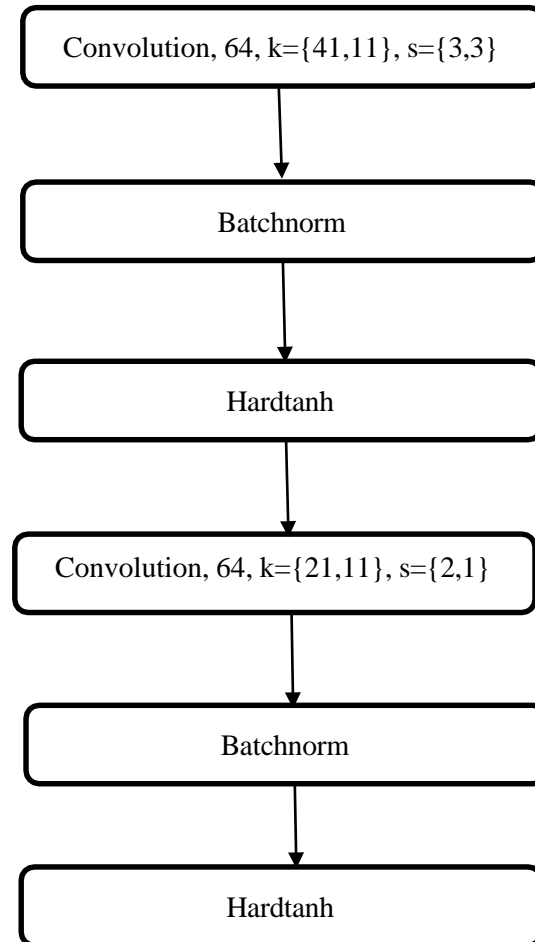


Fig 3.2. Deepspeech2 CNN model [1]

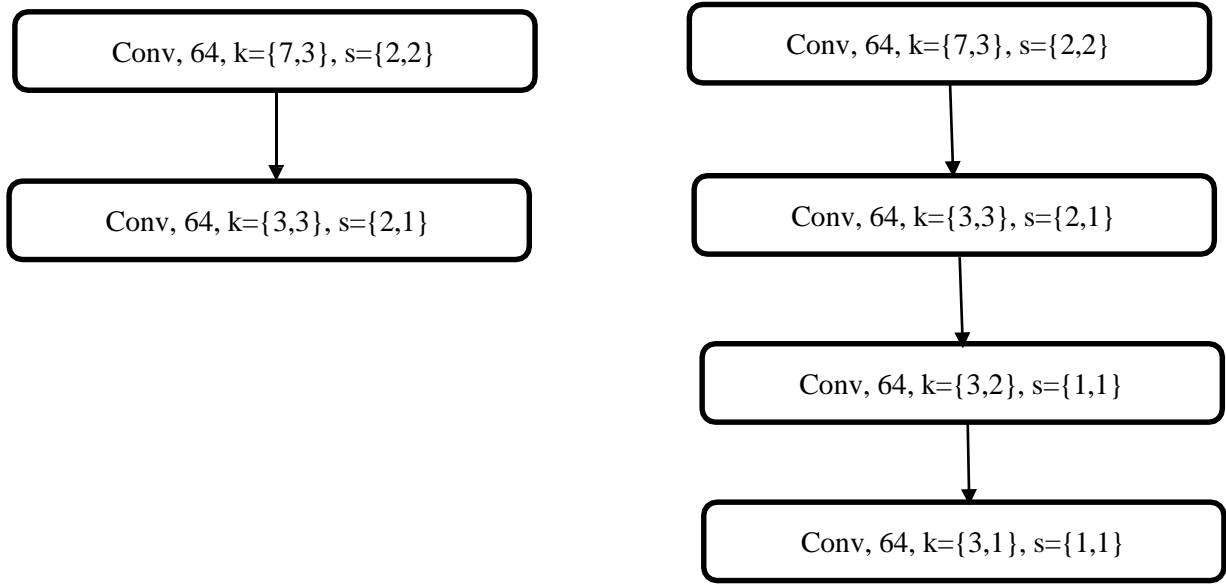


Fig 3.3 Proposed CNN Model

Despite challenges in their real-time implementation, bidirectional recurrent neural systems often achieve superior performance compared to unidirectional models [1]. The quantity of bidirectional layer and the quantity of hidden unit per layers are displayed in Column RNN Config.

### 3.2 TRANSFORMER WITH CTC LOSS

The Transformer is a S2S architecture that efficiently replaces RNN in applications of natural language processing. The development of this technology first focused on neural machine translation (NMT) [1]. I evaluate its performance relative to that of RNN for speech application, such as speech translation (ST) and automatic voice recognition (ASR). The Transformer model necessitates more intricate setups, such as the optimizer, network structure, and data augmentation, compared to typical RNN-based models. This complexity poses a significant difficulty when employing Transformer for speech applications. Our goal is to share our knowledge and experience in using the Transformer model for voice-related activities. We aim to help the community achieve the same impressive outcomes by providing accessible open-source tools and instructions that can be easily replicated.

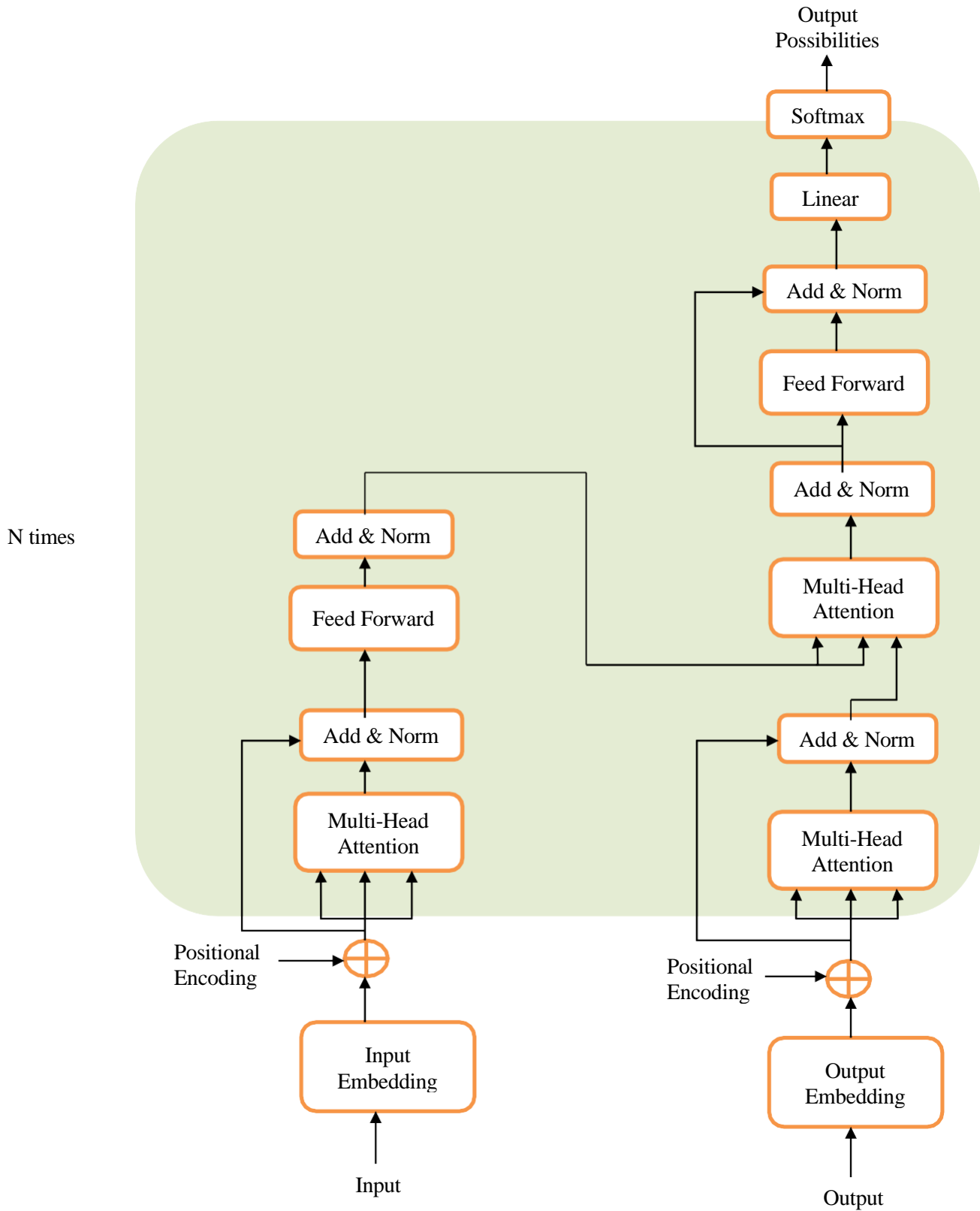


Fig 3.4 Architecture Model of Transformer [9]

### 3.2.1 Encoder-Decoder with Attention

The model's two primary parts are the decoder and the encoder. The decoder produces the target sequence, while the encoder receives the source sequence and outputs a simplified version of it.

Given the encoder's representation and the decoder's previously created tokens, the probability of the series of discrete tokens can be expressed as a sequential product of distribution. A conditional language model describes how the decoder sees the data.

A neural network that can distinguish between the input and output time steps is necessary for both the encoder and the decoder. Additionally, an alternative to recursion must be implemented so that the decoder can depend on certain elements of the encoder's model representation.

### 3.2.2 Multi-Head Attention

At its core, attention is the process of extracting information using a content-based extractor from a set of queries Q, keys K, and values V. The weighted total of the data is returned by the retrieval function, which is predicated on similarities [18] between the queries and the keys:

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T)V$$

The current improvement in dot-product attention is achieved by pre-scaling the queries and introducing sub-space projection for keys, queries, and values into n parallel heads. Attention operations are performed using corresponding heads in this manner. The result is obtained by combining the attention outputs of each head.

Self-attention is a method that gathers information from all time-steps without the need for intermediate transformations. This is different from recurrent connections, which use a single state with a gating mechanism to transmit data, or convolution connections, which combine local states within a limited kernel size.

### 3.3.3 Layer Architecture

Figure 3.3 displays the overall structure of the architecture. The Transformers' encoder and decoder are constructed of stacked layers, each including feed-forward neural networks connected to self-attentional sub-layers. To adjust the encoder for lengthy speech utterances, we follow the reshaping strategy from [19] by merging consecutive frames into a single phase. The input features are subsequently merged with sinusoidal positional encoding [20]. Although the direct addition of auditory characteristics to the positional encoding during training can possibly lead to divergence [19], we managed to circumvent this problem by projecting the concatenated features to a higher dimension (512), similar to the other hidden layers in the model. In the specific context of voice recognition [12], positional encoding is undeniably more effective than learnable positional embeddings. This is due to the fact that speech signals exhibit a greater degree of variability compared to text sequences and can have variable lengths.

## 3.3 ASR training and decoding

With respect to corresponding source  $X$ , the decoder and the CTC module both predict the frame-wise posterior distribution of  $Y$  during ASR training.  $P_{S2S}(\vec{Y}|X)$  and  $P_{CTC}(\vec{Y}|X)$ , using a weighted sum of those

negative Log likelihood values :  $L^{asr} = -\alpha \log_{s2s}(Y/X) - (1-\alpha) \log_{ctc}(Y/X)$ , where  $\alpha$  is a hyperparameter.

Using beam search, which combines the scores of S2S, CTC, and the RNN language model (LM) as follows, the decoder predicts the subsequent token given the speech attribute  $X$  and the prior predicted tokens during the decoding step  $Y = \text{argmax} \{ \lambda \log_{s2s}(Y/X) + (1-\alpha) \log_{ctc}(Y/X) + \alpha \log_{lm}(Y) \}$  where  $Y$  is target sequence hypothesis.

## CHAPTER 4


### EXPERIMENTAL RESULTS

This section is a discussion of the dataset used in experiments and the corresponding quantitative and qualitative results.

#### 4.1 Dataset

The dataset is downloaded from the LJSpeech Dataset. This dataset has a recording of Nearly 13,000 audio files which is in wav files in the /wavs/folder. This audio files is further divided as strings which is given in metadata.csv file. There are:

- ID: Designate the name that matches the .wav file.
- Transcription: Verbal utterances made by the speaker Normalized Transcription: Transcription that includes ordinal numbers, numerical values, and monetary units, all of which are written down in full words.



	file_name	normalized_transcription
0	LJ002-0293	those who could pay nothing went, as a matter ...
1	LJ042-0234	quote, except in the US, the living standard i...
2	LJ049-0191	is properly manned and equipped to carry on ex...

Fig.4.1 speech to text transcription

#### 4.2 Training and Result

For training use of normalized log-spectrogram which is obtained using a window Size of width 200ms and stride 120ms, following by a 160point FFT given as input to the system. Using CTC Loss function the network is trained end to end that is used for character prediction sequentially from input.

Training is performed using A100 Nvidia GPU and subsequently WER and CTC loss are calculated for the RNN and transformer model.

Using spectrogram features the audio is converted into speech to text.

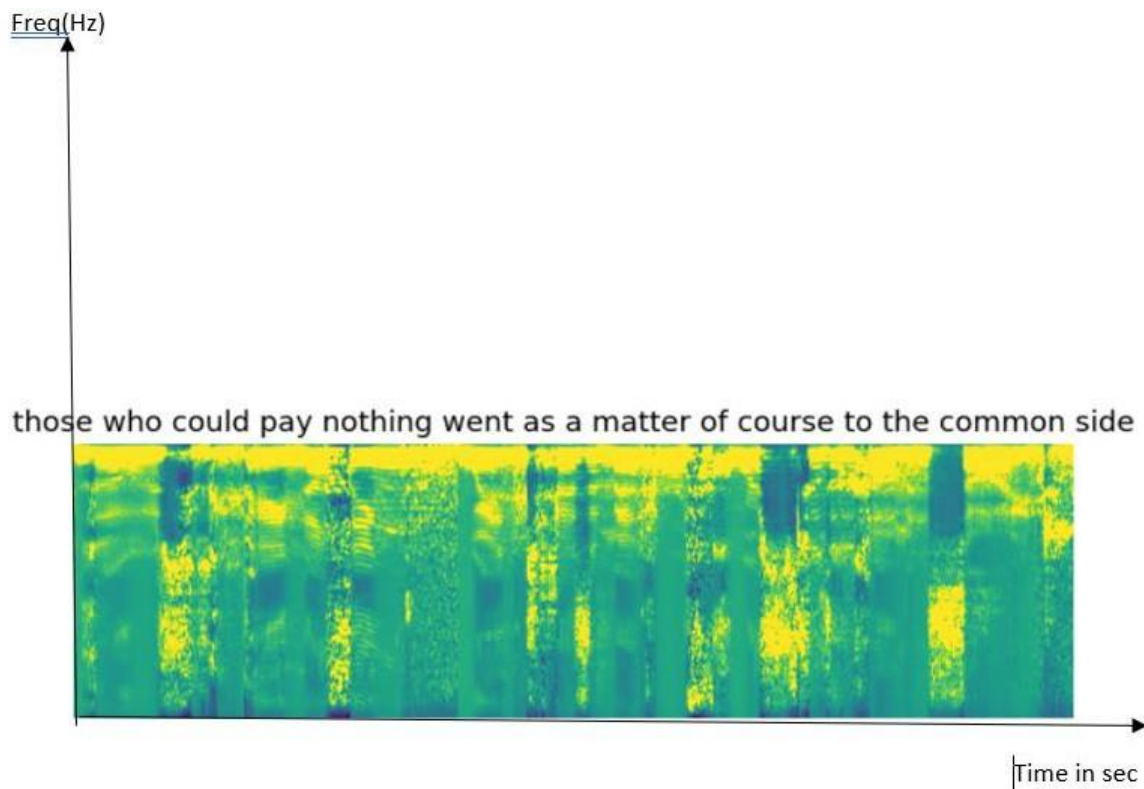


Fig 4.2 audio spectrogram

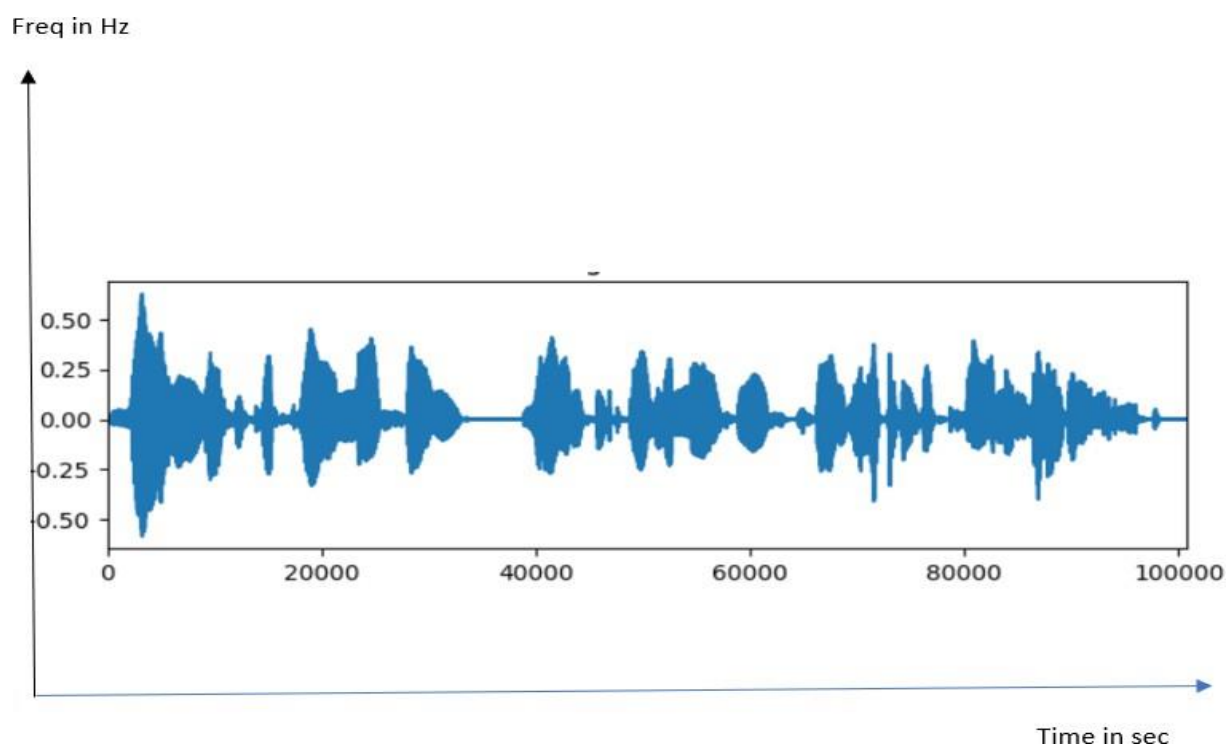


Fig.4.3 signal wave

Model: "DeepSpeech\_2"

Layer (type)	Output Shape	Param #
input (InputLayer)	[(None, None, 193)]	0
expand_dim (Reshape)	(None, None, 193, 1)	0
conv_1 (Conv2D)	(None, None, 97, 32)	14432
conv_1_bn (BatchNormalization)	(None, None, 97, 32)	128
conv_1_relu (ReLU)	(None, None, 97, 32)	0
conv_2 (Conv2D)	(None, None, 49, 32)	236544
conv_2_bn (BatchNormalization)	(None, None, 49, 32)	128
conv_2_relu (ReLU)	(None, None, 49, 32)	0
reshape (Reshape)	(None, None, 1568)	0
bidirectional_1 (Bidirectional)	(None, None, 256)	1304064
dropout (Dropout)	(None, None, 256)	0
bidirectional_2 (Bidirectional)	(None, None, 256)	296448
dropout_1 (Dropout)	(None, None, 256)	0
bidirectional_3 (Bidirectional)	(None, None, 256)	296448
dropout_2 (Dropout)	(None, None, 256)	0
bidirectional_4 (Bidirectional)	(None, None, 256)	296448
dropout_3 (Dropout)	(None, None, 256)	0
bidirectional_5 (Bidirectional)	(None, None, 256)	296448
dense_1 (Dense)	(None, None, 256)	65792
dense_1_relu (ReLU)	(None, None, 256)	0
dropout_4 (Dropout)	(None, None, 256)	0

=====  
 Total params: 2815104 (10.74 MB)  
 Trainable params: 2814976 (10.74 MB)  
 Non-trainable params: 128 (512.00 Byte)

Fig 4.3 Deepspeech 2 Model



### 4.3 RESULT FOR RNN-CNN MODEL

The word error rate is calculated for 50 epochs. Each epoch took around 9-10 minutes on Nvidia GPU A100 on Google Colab-pro. The loss vs epoch curve in Fig. 4.4 and the wer vs epoch curve in Fig. 4.5 shows that the optimal performance is achieved after 35 epochs. This model achieved a word error rate of 22% to 23%.

Word Error Rate: 0.2242

Target : he was himself a prominent member of the low church of austere piety active in all good works  
Prediction: he was himself approminent member of the lo church of austyer pity active in all good works

Target : the nearly indiscriminate admission of visitors although restricted to certain days continued to be an unmixed evil  
Prediction: the nearly indiscriminate admission a visitors although restricted to certain days continued to be and unmixed evil

Target : it was for some time after this a constant practice to go up the chimneys in the hopes of escaping by the flue  
Prediction: it was for sometime after this a counstin practice to gouck the chimties in the hope of escaping by the flu

Target : it is generally supposed that he committed the murder under a sudden access of covetousness and greed  
Prediction: it is generally supposed that he committed the murder under a suden access of covitisness and greed

Target : on november twentytwo however before surrendering possession of the rifle to the fbi laboratory  
Prediction: on november twetdytwo however beforce arendering possession of the rifle to the fbi laboratory

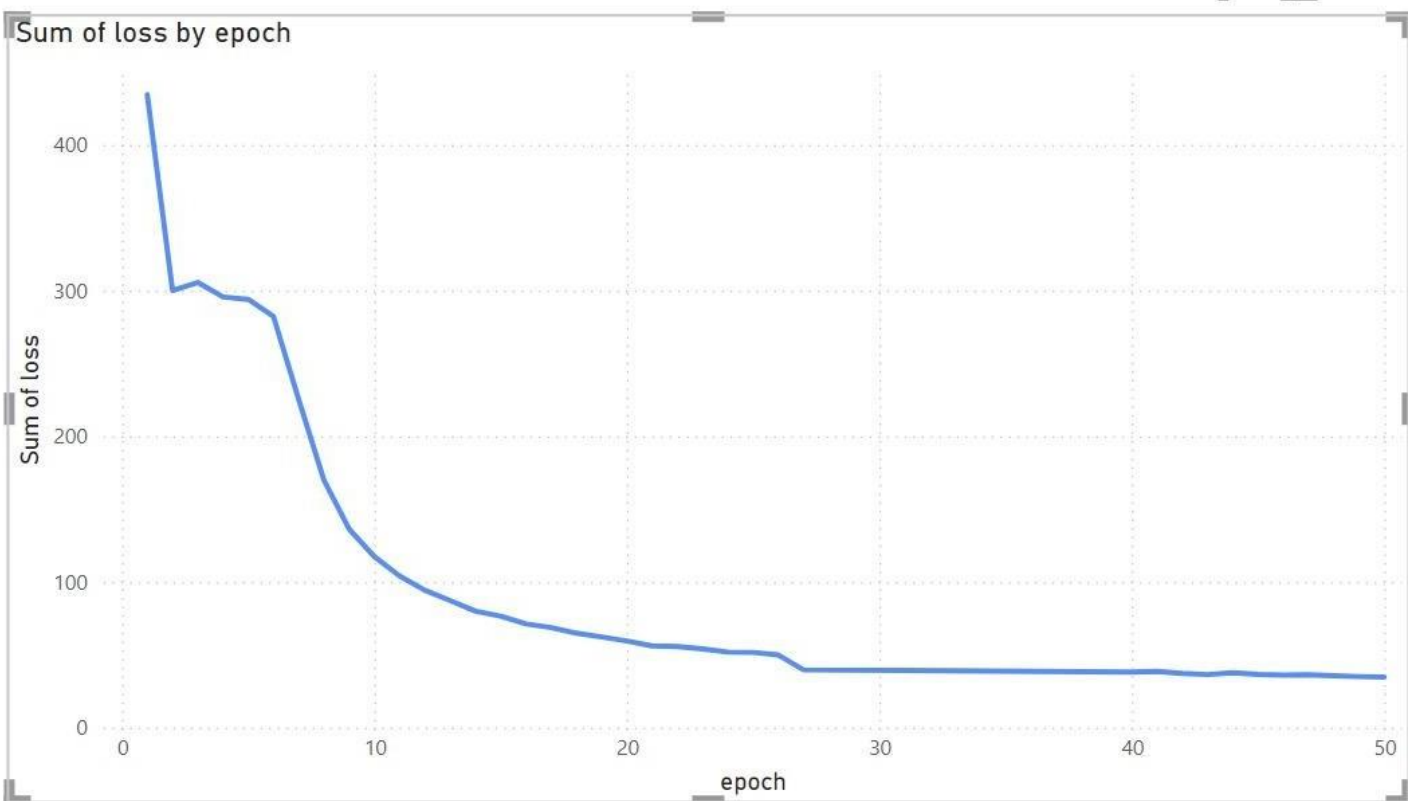


Fig 4.4 Loss Vs epoch graph

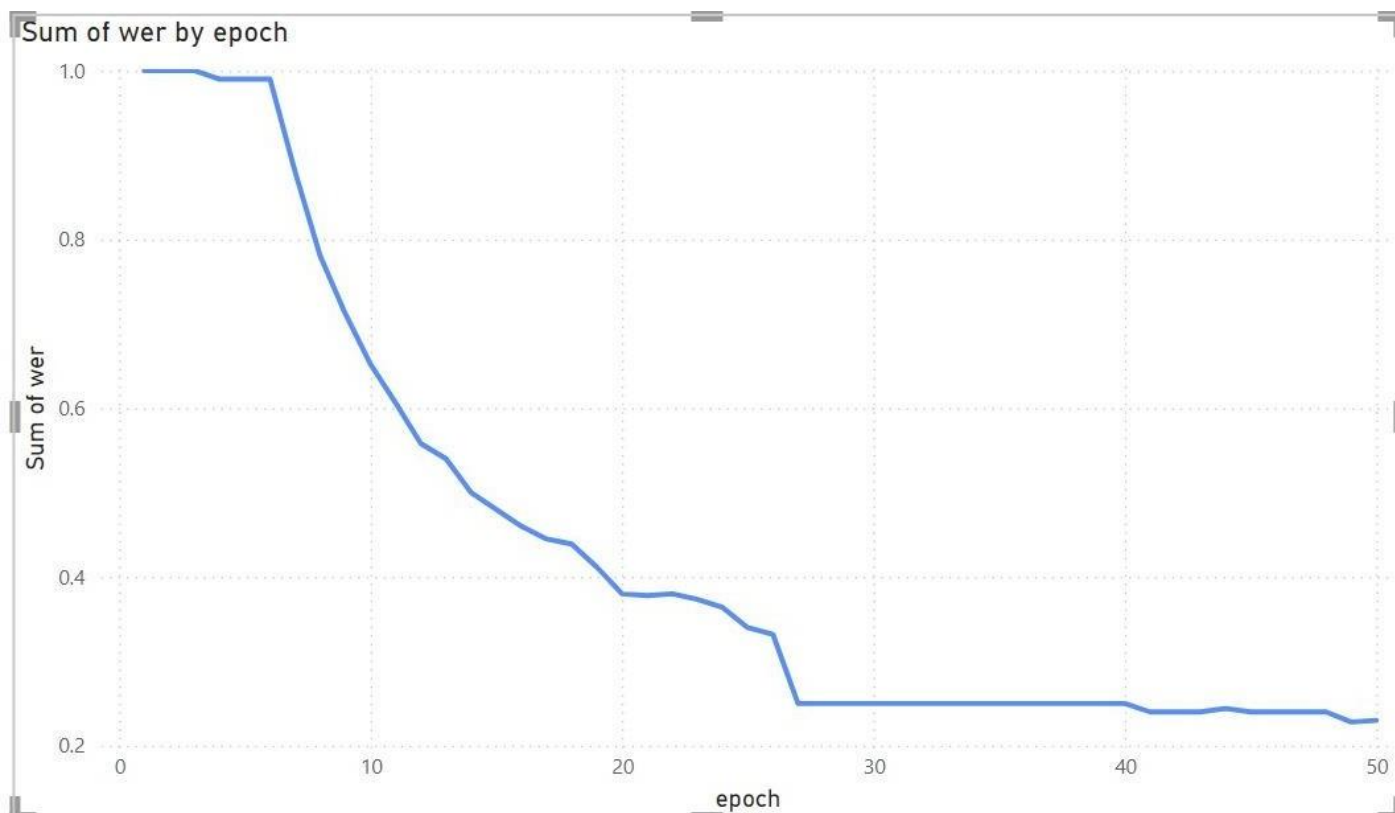


Fig. 4.5 wer vs epoch plot.

## 4.4 Result for the Transformer Model

Epoch 96/100

203/203 [=====] - ETA: 0s - loss: 0.3356 -val\_loss: 0.4927

target: <mr. johnson was the then vice president and his visit took place on april twentythird.>

prediction: <mr. johnson was twe then vice president,and his visit toop a place on aperl twenty third.>

target: <elevated approximately fifteen inches above the back of the front seat>

prediction: <elevated approximated fifteen inches above the back of the front seek,>

The word error rate is calculated for 100 epochs. Each epoch took around 1-2 minutes on Nvidia GPU A100 on Google Colab-pro. The loss vs epoch curve in Fig. 4.6 shows that the optimal performance is achieved after 80 epochs. This model achieved a word error rate of 19% to 20%.

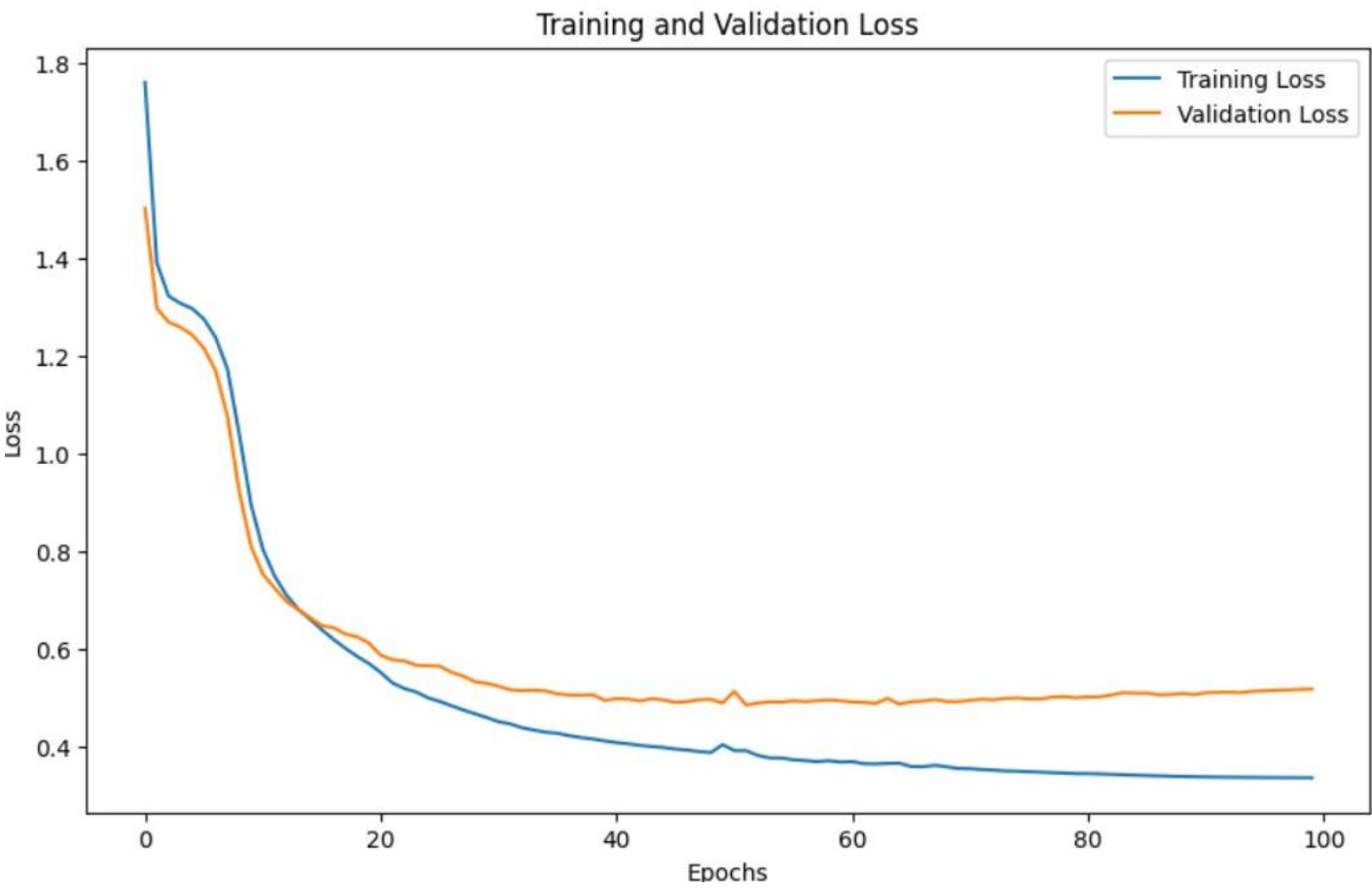


Fig.4.6 Epochs vs loss graph

## **CHAPTER 5**

### **CONCLUSION**

This study examines the effectiveness and suitability of two prominent end-to-end ASR pipelines for the English language. The first pipeline consists of a deep neural network based on CNN-RNN architecture, trained using the CTC loss function. Additionally, a transformer model is also used. The study evaluates and compares these two network configurations with varying complexities. The Transformer-based model has more computational speed compared to the RNN model, however, it is hindered by its inherent complexity. Also, the word alignment in case of long vocabulary sentences needs a greater number of epochs with better tuning of the decoding hyperparameters. There is a significant reduction in wer by 3% to 4% in the case of transformer based model over the considered RNN-CNN model. The suggested network configurations can be extended to other related languages such as Spanish and Dutch, with possibility of similar outcomes due to their shared phonetic space with English. The replicable recipes, pretrained models, and training instructions outlined in this work are expected to expedite research efforts on Transformer based voice applications. Progress in transformer topologies will result in substantial enhancements in transcription precision and effectiveness. These models will improve their capacity to comprehend context by effectively managing long-range dependencies in the speech.

## REFERENCES

- [1] D. Amodei *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. Accessed: May 23, 2024. [Online]. Available: <http://proceedings.mlr.press/v48/amodei16.html?ref=https://codemonkey.link>
- [2] G. Hinton *et al.*, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, 2012.
- [3] D. Povey *et al.*, “Purely sequence-trained neural networks for ASR based on lattice-free MMI,” in *Interspeech*, 2016, pp. 2751–2755. Accessed: May 23, 2024. [Online]. Available: [https://www.isca-archive.org/interspeech\\_2016/povey16\\_interspeech.pdf](https://www.isca-archive.org/interspeech_2016/povey16_interspeech.pdf)
- [4] T. Nakatani, “Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration,” in *proc. INTERSPEECH*, 2019, pp. 1408–1412. Accessed: May 23, 2024. [Online]. Available: [https://www.isca-archive.org/interspeech\\_2019/karita19\\_interspeech.pdf](https://www.isca-archive.org/interspeech_2019/karita19_interspeech.pdf)
- [5] A. Graves, “Sequence Transduction with Recurrent Neural Networks.” arXiv, Nov. 14, 2012. Accessed: May 23, 2024. [Online]. Available: <http://arxiv.org/abs/1211.3711>
- [6] J. Li, R. Zhao, H. Hu, and Y. Gong, “Improving RNN transducer modeling for end-to-end speech recognition,” in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2019, pp. 114–121. Accessed: May 23, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/9003906/?casa\\_token=I5RhET-QZtQAAAAA:\\_FiCT-FCjMJk9odfT4wWEayanir-eL6WvM4Eeah9dUZ2UJTbmd4THofCvnXO33RFFzOEJ-tNtQ](https://ieeexplore.ieee.org/abstract/document/9003906/?casa_token=I5RhET-QZtQAAAAA:_FiCT-FCjMJk9odfT4wWEayanir-eL6WvM4Eeah9dUZ2UJTbmd4THofCvnXO33RFFzOEJ-tNtQ)
- [7] J. Schalkwyk, “An all-neural on-device speech recognizer,” *Latest News Google AI*, 2019.
- [8] N. Moritz, T. Hori, and J. Le Roux, “Unidirectional Neural Network Architectures for End-to-End Automatic Speech Recognition,” in *INTERSPEECH*, 2019, pp. 76–80. Accessed: May 25, 2024. [Online]. Available: <https://merl.com/publications/docs/TR2019-098.pdf>
- [9] S. Karita *et al.*, “A comparative study on transformer vs rnn in speech applications,” in *2019 IEEE automatic speech recognition and understanding workshop (ASRU)*, IEEE, 2019, pp. 449–456. Accessed: May 25, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/9003750/?casa\\_token=xZGgwhc3AW8AAAAA:9EGwAg3yUBNSmIGyg-SP8Iyy3Bm6a\\_WjXT10UbWkzlpCRM9fF9JPbVX3Jbm4XfLM9gKPMYg9pWF4](https://ieeexplore.ieee.org/abstract/document/9003750/?casa_token=xZGgwhc3AW8AAAAA:9EGwAg3yUBNSmIGyg-SP8Iyy3Bm6a_WjXT10UbWkzlpCRM9fF9JPbVX3Jbm4XfLM9gKPMYg9pWF4)
- [10] N. Moritz, T. Hori, and J. Le Roux, “Triggered attention for end-to-end speech recognition,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 5666–5670. Accessed: May 25, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/8683510/?casa\\_token=Lmn9HhFFvrAAAAAA:tapIFj9ceNwKfJLo7huFu14bmZbWzItGszAIXyc81yuaxJskXrzOQbU6SRsv1leDHIaryyjKp2yN](https://ieeexplore.ieee.org/abstract/document/8683510/?casa_token=Lmn9HhFFvrAAAAAA:tapIFj9ceNwKfJLo7huFu14bmZbWzItGszAIXyc81yuaxJskXrzOQbU6SRsv1leDHIaryyjKp2yN)
- [11] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning - ICML '06*, Pittsburgh, Pennsylvania: ACM Press, 2006, pp. 369–376. doi: 10.1145/1143844.1143891.
- [12] A. Hannun *et al.*, “Deep Speech: Scaling up end-to-end speech recognition.” arXiv, Dec. 19, 2014. Accessed: May 26, 2024. [Online]. Available: <http://arxiv.org/abs/1412.5567>
- [13] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” *Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, Accessed: May 26, 2024. [Online]. Available: <https://proceedings.neurips.cc/paper/2015/hash/1068c6e4c8051cfd4e9ea8072e3189e2-Abstract.html>
- [14] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2016, pp. 4960–4964. Accessed: May 26, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/7472621/?casa\\_token=aP31rW50BI0AAAAA:YHMhl6jLAbjHM-aP-ci-4dIUf9Y51jQ75KT\\_Ry2QcUmSABPaclyG3mAVNWyoQEOBC9O6YlnpeqR3](https://ieeexplore.ieee.org/abstract/document/7472621/?casa_token=aP31rW50BI0AAAAA:YHMhl6jLAbjHM-aP-ci-4dIUf9Y51jQ75KT_Ry2QcUmSABPaclyG3mAVNWyoQEOBC9O6YlnpeqR3)
- [15] K. Rao, H. Sak, and R. Prabhavalkar, “Exploring architectures, data and units for streaming end-to-end

- speech recognition with rnn-transducer,” in *2017 IEEE automatic speech recognition and understanding workshop (ASRU)*, IEEE, 2017, pp. 193–199. Accessed: May 26, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/8268935/?casa\\_token=vbEYm6uHySwAAAAA:caotE7pvJvvaHkkz3hZqDIAz\\_o73NC5SOxD3hfWxAzT9Edh5Kl\\_vDA22z1E6OQilCEy\\_LC7Cd4R8](https://ieeexplore.ieee.org/abstract/document/8268935/?casa_token=vbEYm6uHySwAAAAA:caotE7pvJvvaHkkz3hZqDIAz_o73NC5SOxD3hfWxAzT9Edh5Kl_vDA22z1E6OQilCEy_LC7Cd4R8)
- [16] K. Rao, H. Sak, and R. Prabhavalkar, “Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer,” in *2017 IEEE automatic speech recognition and understanding workshop (ASRU)*, IEEE, 2017, pp. 193–199. Accessed: May 26, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/8268935/?casa\\_token=vbEYm6uHySwAAAAA:caotE7pvJvvaHkkz3hZqDIAz\\_o73NC5SOxD3hfWxAzT9Edh5Kl\\_vDA22z1E6OQilCEy\\_LC7Cd4R8](https://ieeexplore.ieee.org/abstract/document/8268935/?casa_token=vbEYm6uHySwAAAAA:caotE7pvJvvaHkkz3hZqDIAz_o73NC5SOxD3hfWxAzT9Edh5Kl_vDA22z1E6OQilCEy_LC7Cd4R8)
- [17] Y. Miao, M. Gowayyed, and F. Metze, “EESSEN: End-to-end speech recognition using deep RNN models and WFST-based decoding,” in *2015 IEEE workshop on automatic speech recognition and understanding (ASRU)*, IEEE, 2015, pp. 167–174. Accessed: May 26, 2024. [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/7404790/?casa\\_token=28Smq1\\_EMMsAAAAA:eSgPg0S5-FdccLKuJmekDb9b3wh\\_kHWxaosBoxYBJKzLx44KP3BM6N7IoyyQWUa6EnOMDzo](https://ieeexplore.ieee.org/abstract/document/7404790/?casa_token=28Smq1_EMMsAAAAA:eSgPg0S5-FdccLKuJmekDb9b3wh_kHWxaosBoxYBJKzLx44KP3BM6N7IoyyQWUa6EnOMDzo)
- [18] M.-T. Luong, H. Pham, and C. D. Manning, “Effective Approaches to Attention-based Neural Machine Translation.” arXiv, Sep. 20, 2015. Accessed: May 27, 2024. [Online]. Available: <http://arxiv.org/abs/1508.04025>
- [19] M. Sperber, J. Niehues, G. Neubig, S. Stüker, and A. Waibel, “Self-Attentional Acoustic Models.” arXiv, Jun. 18, 2018. Accessed: May 27, 2024. [Online]. Available: <http://arxiv.org/abs/1803.09519>
- [20] A. Vaswani *et al.*, “Attention is all you need,” *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, Accessed: May 27, 2024. [Online]. Available: <https://proceedings.neurips.cc/paper/7181-attention-is-all>

PAPER NAME

**thesis copy.docx**

AUTHOR

**rupesh e2e ASR**

WORD COUNT

**5215 Words**

CHARACTER COUNT

**31704 Characters**

PAGE COUNT

**30 Pages**

FILE SIZE

**958.3KB**

SUBMISSION DATE

**May 30, 2024 7:53 PM GMT+5:30**

REPORT DATE

**May 30, 2024 7:53 PM GMT+5:30**

### ● 16% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 11% Internet database
- 10% Publications database
- Crossref database
- Crossref Posted Content database
- 8% Submitted Works database

### ● Excluded from Similarity Report

- Bibliographic material
- Manually excluded text blocks