LOW RESOURCE AUTOMATIC SPEECH RECOGNITION

PROJECT - PHASE II REPORT

Submitted by

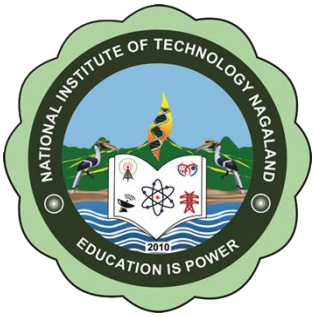
GAALI VAMSHI (2019104196)
KOKKANTI SANDEEP REDDY (2019104201)

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY NAGALAND
DIMAPUR-797 103
May 2023

LOW RESOURCE AUTOMATIC SPEECH RECOGNITION

PROJECT – PHASE II REPORT

Submitted by

GAALI VAMSHI (2019104196)
KOKKANTI SANDEEP REDDY (2019104201)

in partial fulfilment for the award of the degree of

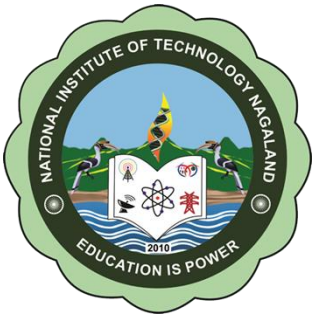
BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

Under the supervision of

Dr. Madhusudan Singh



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY NAGALAND
DIMAPUR 797 103
May 2023

राष्ट्रीय प्रौद्योगिकी संस्थान नागालैंड
NATIONAL INSTITUTE OF TECHNOLOGY NAGALAND
DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING
Chumukedima, Dimapur – 797 103, Nagaland

BONAFIDE CERTIFICATE

Certified that this Project titled "**Low Resource Automatic Speech Recognition**" is the bonafide work of Kokkanti Sandeep Reddy (2019104201) and Gaali Vamshi (2019104196) who carried out the work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation based on which a degree or award was conferred on an earlier occasion on this or any other student.

Dr. Madhusudan Singh
Project Supervisor
Assistant Professor
Department of Electronics and Communication Engineering
NIT Nagaland

Dr. Madhusudan Singh
Project Coordinator
Assistant Professor
Department of Electronics and Communication Engineering
NIT Nagaland

Dr. Jay Chandra Dhar
Head of the Department
Assistant Professor
Department of Electronics and Communication Engineering
NIT Nagaland

External Examiner

ACKNOWLEDGEMENT

We place on record our deep sense of gratitude to our Guide **Dr. Madhusudan Singh**, Asst. Professor, NIT Nagaland for his supervision, valuable guidance and moral support leading to the successful completion of the work. Without his continuous encouragement and involvement, this project would not have been a reality.

We wish to thank **Dr. Jay Chandra Dhar**, Head of the Department for his continuous support. We also thank all our friends and seniors who have developed us to gain a sense of dutifulness, perfection and sincerity in the effort.

Gaali Vamshi (2019104196)

Kokkanti Sandeep Reddy (2019104201)

ABSTRACT

Deep learning techniques have made substantial advancements in speech recognition. As a result, businesses are focusing more and more on utilizing speech recognition in their voice command systems in order to benefit from these innovative advancements.

The most effective configuration of these arrangements in various circumstances is currently unknown, although data preprocessing, the size of the dataset, and the architecture of the deep learning model may all have a significant impact on the accuracy of speech recognition.

Automatic speech recognition technologies require a large amount of annotated data for a system to work reasonably well. However, for many languages in the world, not enough speech data is available, or these lack the annotations needed to train an ASR system. In fact, it is estimated that for only about 1% of the world languages the minimum amount of data that is needed to train an ASR is available. In this we propose a model to build an ASR system for a Low Resource Language.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ASR - Automatic Speech Recognition

ReLU - Rectified Linear Unit

CNN - Convolutional Neural Network

DNN - Deep Neural Network

RNN - Recurrent Neural Network

LSTM - Long Short-Term Memory

DFT - Discrete Fourier Transform

FFT - Fast Fourier Transform

STFT - Short Time Fourier Transform

CONV - Convolutional Layer

POOL - Pooling Layer

CER - Character Error Rate

CHAPTER 1

INTRODUCTION

1.1 GENERAL

The automatic speech recognition (ASR) technology has advanced significantly in recent years thanks to advancements in computer technology and the development of machine learning theory. As a result, ASR is now a mature technology set that has seen extensive use and has had great success in interface applications.

Three modules, comprising acoustic models, language models, and pronunciation dictionaries, make up a typical ASR system. Large vocabulary continuous speech recognition technology is become increasingly useful.

However, as economic globalization has progressed, speech recognition is no longer just available for widely spoken languages like Chinese and English. More and more thought is being given to how to best utilize the limited resources of speech data to create a high-performance speech recognition system.

Significant data resources and linguistic expertise are frequently needed for speech recognition. However, for low-resource languages, there are limitations on the development of low-resource ASR due to a lack of linguistic expertise as well as a lack of significant quantities of speech data for speech acoustic model training.

Recent research in speech recognition technology has focused on how robust speech recognition systems are against linguistic variance.

The most important prerequisite for creating effective speech recognition technology is to create a system that can converse with humans in any language like any other human.

With a population of more than a billion people, India has a very diverse linguistic community. As a result, it provides a strong field of study for speech recognition systems suited to certain languages [5].

1.2 OBJECTIVES

1. To develop an ASR system for low resource language.
    - To develop a base line ASR system.
    - To develop a Low Resource ASR system using transfer learning technique.

1.3 ORGANIZATION OF THE REPORT

Chapter 1 Explains the introduction and objective of the project.

Chapter 2 Gives brief data of Literature Review.

Chapter 3 Gives an overview of ASR system.

Chapter 4 Explains different model architectures used for ASR system.

Chapter 5 Consists of results and discussion.

Chapter 6 Concludes paper and provides information on future scope.

CHAPTER 2

STATE OF THE ART

2.1. GENERAL

Literature review is the summary of the source related to the research. In this chapter all the work and inferences drawn from different sources on Low Resource Automatic speech recognition system has been highlighted.

2.2. LITERATURE REVIEW

1. Shashank Holla S (2022), “End-to-End Speech Recognition for Low Resource Language Sanskrit using Self-Supervised Learning”. In this paper an ASR system for a low resource language using the self-supervised learning model is proposed. The model is trained on nearly 10,000 hrs. of un labelled audio data and fine-tuned on 78 hrs. labelled Sanskrit audio data[13].
2. Ethan Morris, Rochester Institute of Technology RIT Scholar Works (2021), “Automatic Speech Recognition for Low-Resource and Morphologically Complex Languages”.In this paper they Proposed a fully convolutional neural network model that allows for the skipping of the time-consuming transfer learning stage, as well as general architecture adjustments which better suit an array of various morphological languages[10].
3. Raymond Ptucha, Robert Jimerson (2019), “SYNTHETIC DATA AUGMENTATION FOR IMPROVING LOW-RESOURCE ASR”.In this paper they proposed that the Transfer learning and data augmentation independently contribute to meaningful reductions in word error rate[4].
4. Mannon Ochilov, Ilyos Xujayorov (2019). “Image Approach to Speech Recognition on CNN”. In this paper, the classification of speech sounds based on spectrogram images has been performed by using CNN.As the results of experiments show that based on this approach, it is possible to achieve good results by classifying the sounds.

The extraction of features from the spectrogram is performed automatically in deep learning. In machine learning algorithms, the process of selecting and extracting characters is performed manually [12].

5. Sayan Mandal, Sarthak Yadav and Atul Rai (2020). “End-to-End Bengali Speech Recognition”. In this paper, an end-to-end automatic speech recognition pipeline comprising of CNN-RNN based deep neural network trained using the CTC loss function for the Bengali language is developed [1].
6. Sameer Bansal, Herman Kamper, Karen Livescu, Adam Lopez, Sharon Goldwater (2019). “Pre-training on high-resource speech recognition improves low-resource speech-to-text translation”. In this paper ASR system for low resource language is developed by transfer learning technique. By using transfer learning, we can get better results than directly training the model with low resource [6].
7. K. -H. Lu and K. -Y. Chen, "A Context-Aware Knowledge Transferring Strategy for CTC-Based ASR," (2022). In this the usage of CTC loss increases the efficiency of ASR system rather than other losses as CTC overcomes the alignment problem [14].

CHAPTER 3

ASR SYSTEM

3.1. ASR MODEL

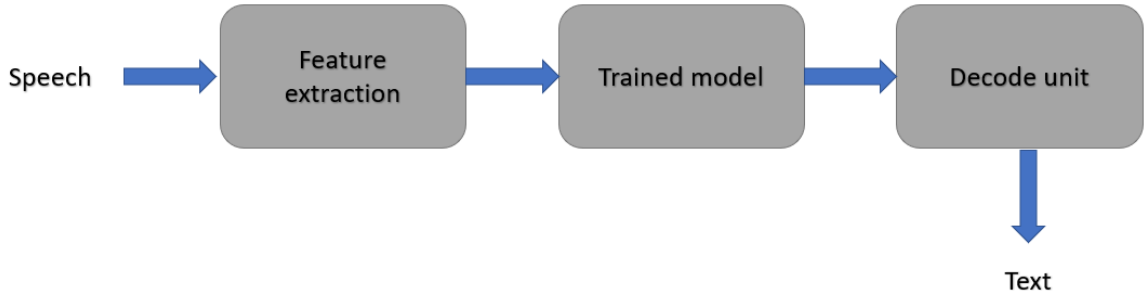


Figure 3.1: ASR Model Block Diagram

3.2. FEATURE EXTRACTION

The feature extraction of speech signal consists of three stages, they are:

- Converting Analog signal to Digital signal.
- Spectrogram conversion
- Mel-spectrogram conversion

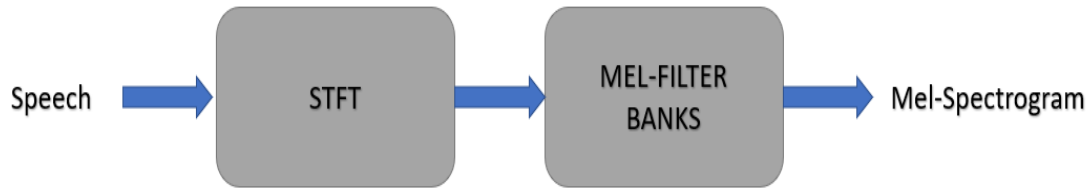


Figure 3.2: Feature Extraction Block Diagram

3.2.1. CONVERTING ANALOG SIGNAL TO DIGITAL SIGNAL: Sampling is a process by which an analog signal is converted into series of samples. This is basically done to convert the continuous amplitude/time signals to discrete amplitude/time signals.

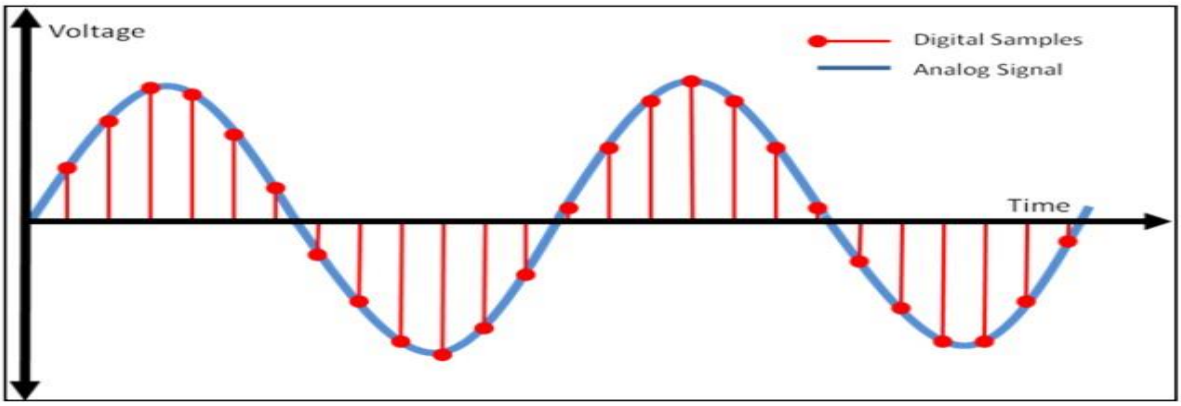


Figure 3.3: Data Sampling

3.2.2. SPECTROGRAM CONVERSION:

The purpose of feature extraction is to illustrate a speech signal by a predetermined number of components of the signal. It employs to converts digital speech signal to sets of feature vectors. These feature vectors contain the essential characteristics of the speaker's voice. This is usually called the front-end signal-processing. With front end being the initial element in the sequence, the quality of the subsequent features is significantly affected by the quality of the front end.

The acoustic signal analysis method can be processed in the time domain or the frequency domain, but they have their limitations: the time domain analysis is not sensitive to the frequency, and the frequency domain analysis cannot change over the time. Therefore, it is necessary to combine both analyses in the time domain and the frequency domain

The STFT is the most commonly used for the time-frequency analysis, it is based on the Fourier transformation which is a method to analyze the signal to be used to move the window function over the time domain to calculate the power spectral density function at the different moments.

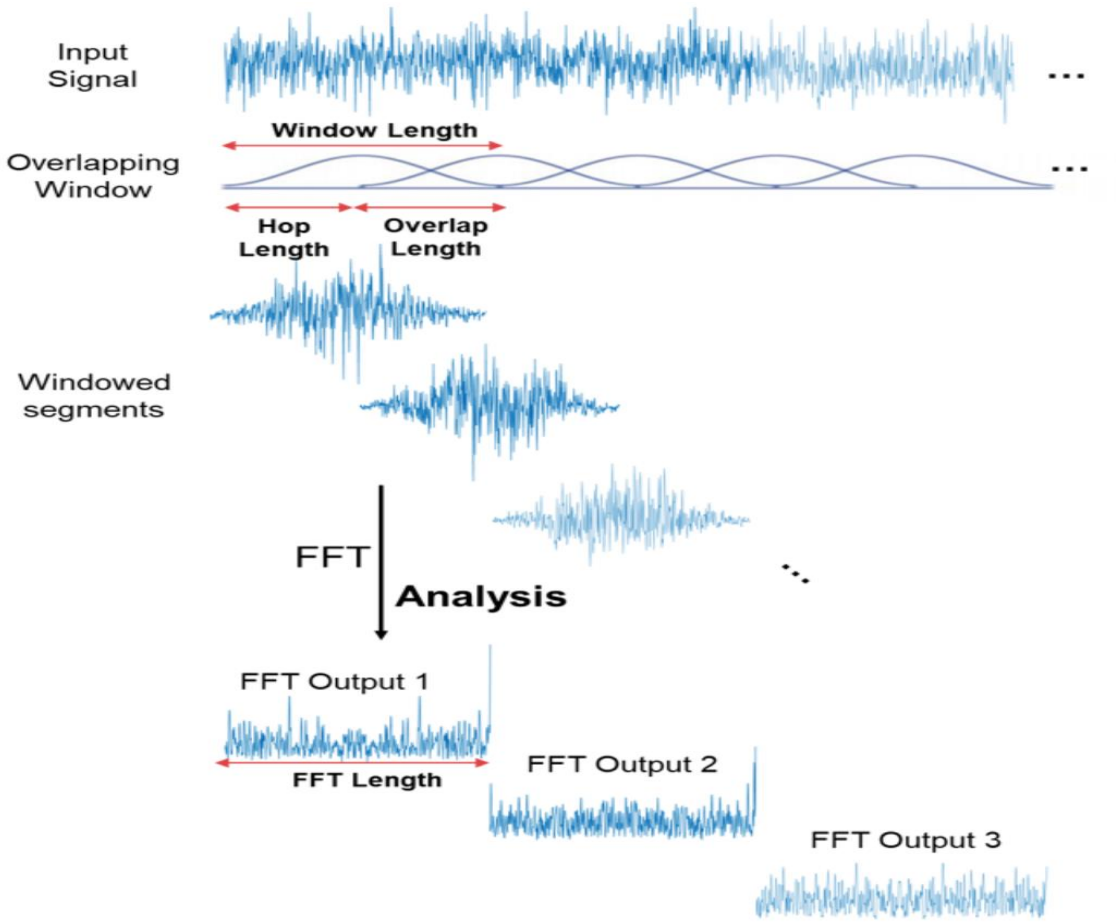


Figure 3.4: STFT Representation

SPECTROGRAM: A spectrogram is an image that asserts that signal power and intensity are time-dependent. It might be presented in a two-dimensional graph. The x-axis shows the time and Y-axis shows frequency. Frequency characteristics of actual amplitude of time are determined by color intensity.

By windowing and taking the discrete Fourier transform (DFT) of each window, we obtain the short-time Fourier transform (STFT) of the signal [12].

Figure 3.5: Spectrogram Block diagram

3.2.3. MEL-SPECTROGRAM CONVERSION:

MEL SCALE: Humans do not hear frequencies on a linear scale, according to studies**.** Lower frequencies are easier for us to distinguish from higher frequencies. For example, even though the distance between the two pairs is the same, we can easily distinguish between 500 and 1000 Hz but will find it difficult to distinguish between 10,000 and 10,500 Hz [12].

The approximation of Mel from physical frequency can be expressed as,

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

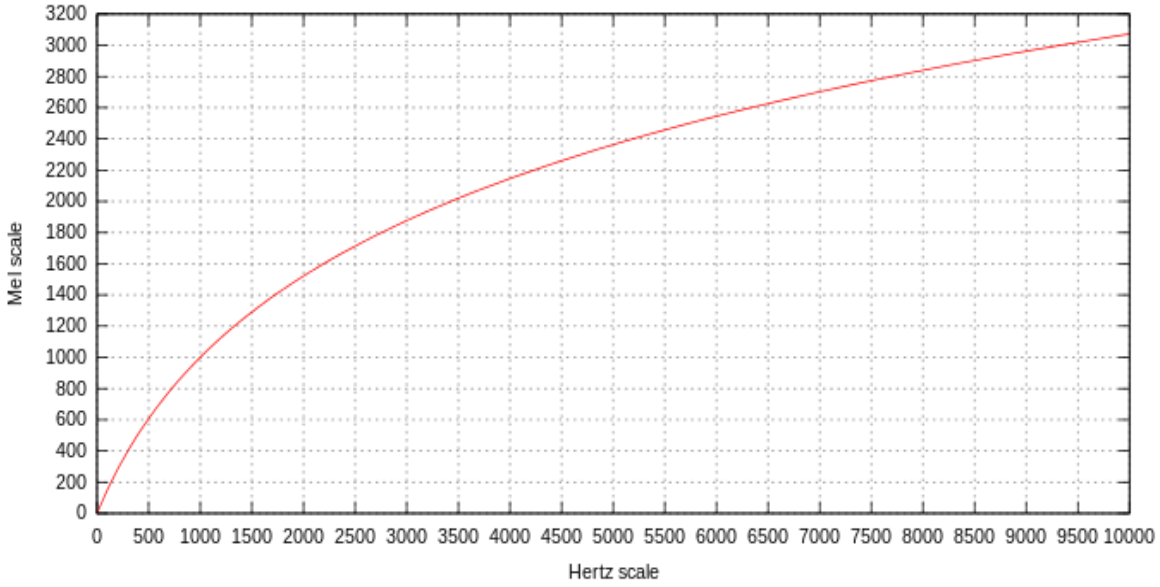


Figure 3.6: Mel Scale Representation

MEL FILTER BANK: Mel Filter Banks is a triangular filter bank that works similar to the human ears perception of sound which is more discriminative at lower frequencies and less discriminative at higher frequencies.

Mel Filter Banks are used to provide a better resolution at low frequencies and less resolution at high frequencies. It captures the energy at each critical frequency band and gives approximates the spectrum shape.

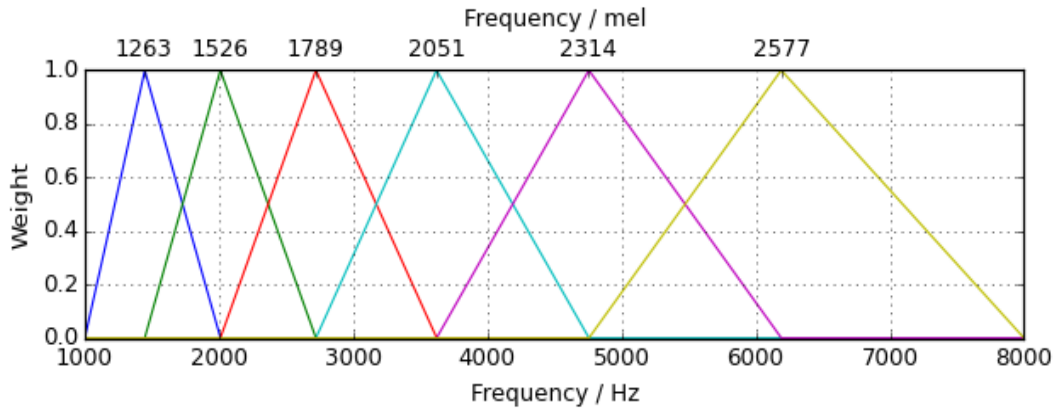


Figure 3.7: Mel Filter Bank Representation

MEL SPECTROGRAM: A Mel spectrogram is a spectrogram where the frequencies are converted to the Mel scale [12].

Figure 3.8: Mel Spectrogram Block Diagram

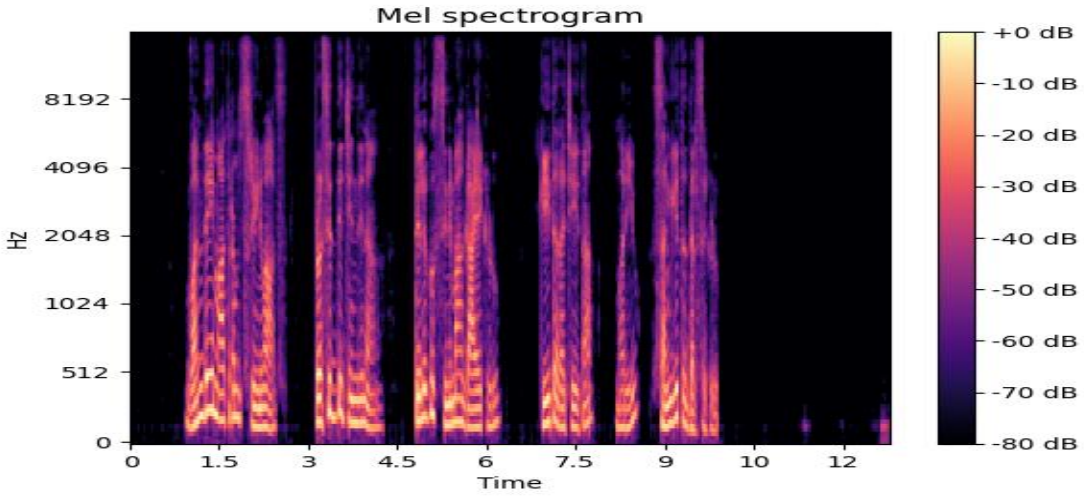


Figure 3.9: Mel Spectrogram

3.3. DEEP NEURAL NETWORK

Deep neural networks are widely used in business applications as a classification and recognition technique. Better predictions on the non-linear data can be made by stacking hidden layers over enormous amounts of data because there is no sense of spatial representation. To map the weights to a standard state, each hidden layer employs an activation function, frequently a Rectified Linear Unit (ReLU). A SoftMax nonlinearity is used for multiclass classification, such as character prediction in an ASR system.

3.3.1. CONVOLUTIONAL NEURAL NETWORK: A Convolutional Neural Network, also known as CNN or ConvNet, is a class of neural networks that specializes in processing data that has a grid-like topology, such as an image. It generally consists of following layers [12].

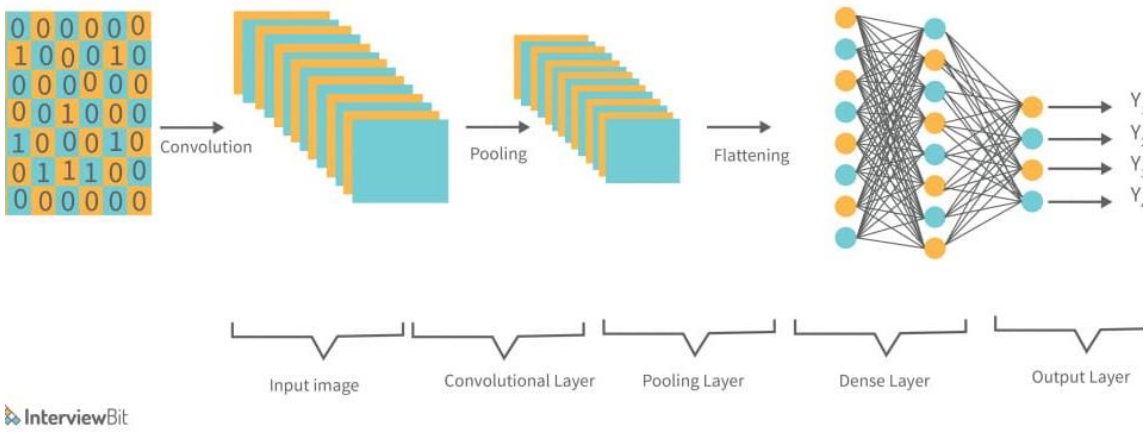


Figure 3.10: CNN Representation

TYPES OF LAYERS:

CONVOLUTIONAL LAYER (CONV): The convolution layer (CONV) uses filters that perform convolution operations as it is scanning the input I with respect to its dimensions. Its hyperparameters include the filter size F and stride S. The resulting output O is called feature map or activation map [9].

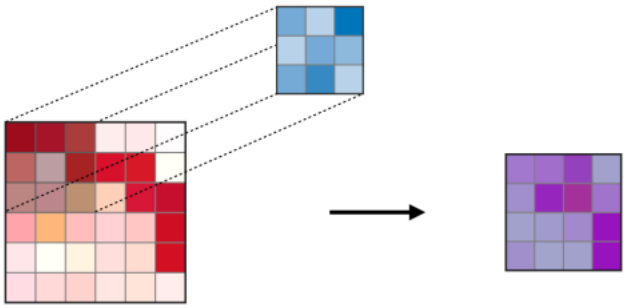


Figure 3.11: CONV Representation

POOLING (POOL) : The pooling layer (POOL) is a down sampling operation, typically applied after a convolution layer, which does some spatial invariance.

In particular, max and average pooling are special kinds of pooling where the maximum and average value is taken, respectively [9].

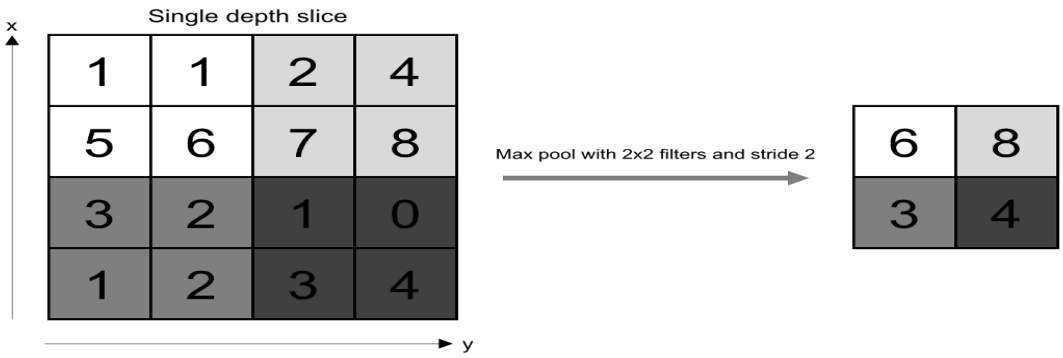


Figure 3.12: POOL Representation

DENSE LAYER: A Dense Layer, also known as a fully connected layer, is a layer in a neural network in which each neuron is connected to every neuron in the previous layer. In a Dense Layer, each neuron performs a linear transformation followed by a non-linear activation function. The output of a Dense Layer is a tensor with the same number of dimensions as the input tensor, but potentially a different size.

TIME DISTRIBUTED LAYER: A Time Distributed Dense Layer is a type of layer used in neural networks for sequence prediction tasks such as speech recognition,

language modeling, and video analysis. In a Time Distributed Dense Layer, the same dense layer is applied to every time step of a sequence independently. This means that each time step of the input sequence is transformed by the same set of weights and biases, resulting in a new sequence of the same length. The output of a Time Distributed Dense Layer is a tensor with the same shape as the input tensor, with each time step transformed by the same set of weights.

3.3.2. RECCURENT NEURAL NETWORKS: The standard recurrent neural network (RNN) model has a loop the hidden unit as shown in Figure 3.13. It has three types of layers: the input layer x, the hidden layer h, and the output layer o. If we unfold this loop, the standard RNN can be considered as copying the same structure multiple times, and the state h of each copy is taken as an input to its successor. Denoting the input layer, hidden layer and output layer at time t as x(t), h(t) and o(t), respectively[1].

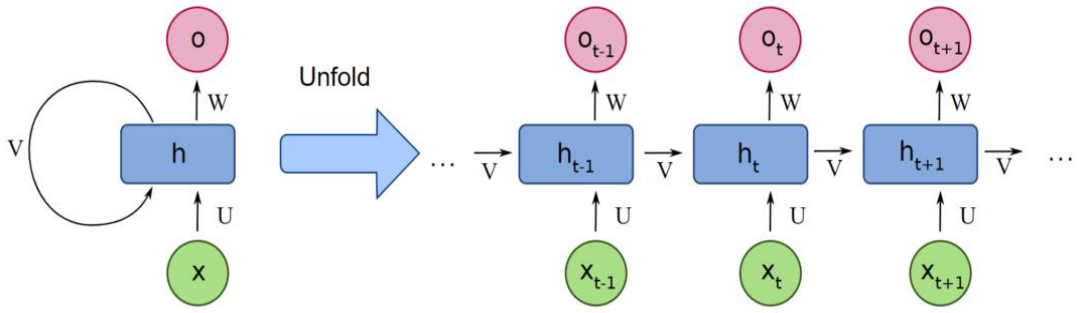


Figure 3.13: Standard RNN

The result of many applications, such as speech recognition and handwriting recognition, depends on the entire input sequence. A bidirectional recurrent neural network that combines two RNNs—one that starts at the beginning of the sequence and goes forward through time, the other that starts at the conclusion of the sequence and moves backward through time—was developed by Schuster et al to satisfy this criterion. The general structure of a bidirectional RNN is shown in Figure 3.14. h(t) stands for the state of the forward-RNN moving forward through time, and h'(t) is used to represent the state of the backward-RNN moving backward through time (t).

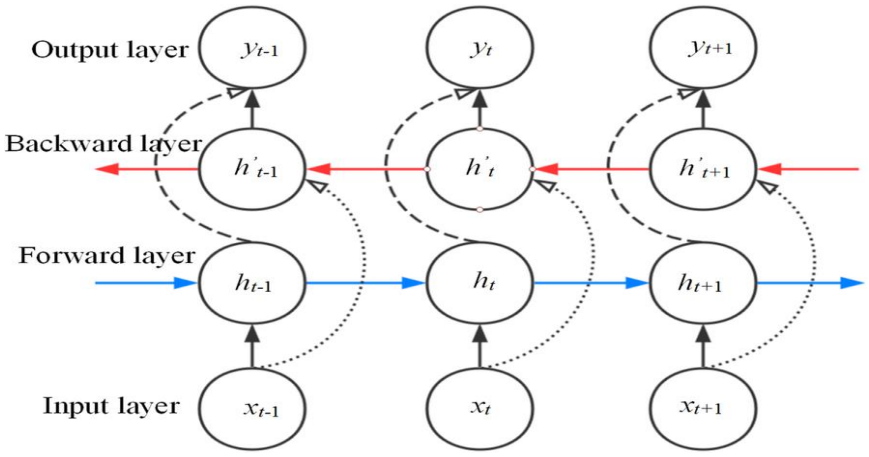


Figure 3.14: Bidirectional RNN

3.3.3. LONG SHORT-TERM MEMORY(LSTM): Due to the gradient vanishing and exploding problem, training standard RNNs is difficult. A long short-term memory (LSTM) model was developed by Hocreiter and Schmidhuber to address this problem. In particular, LSTM uses three gates to regulate whether to forget the current cell (forget gate f), receive its input (input gate I and output the new cell value (Figure 3.15), replacing the hidden layer of the regular RNNs with a memory cell c. (output gate o). Each of these gates only affects one layer. LSTM maintains the value of the relevant layer if the gate is set to 1, and it shrinks this value to zero if the gate is set to 0 [6].

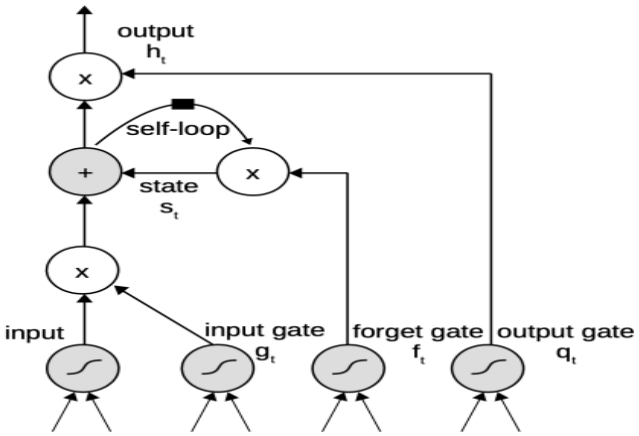


Figure 3.15: LSTM cell Block

3.4. ENCODE AND DECODE UNITS

Working on a Speech Recognition project can be a tedious task, in particular when the data is in textual format and the models require numerical values. To overcome this problem, we can create a dictionary that can map alphabets to numerical values.

To train the model we convert the text to integers using the character map and the output of the model will be decoded to text using sane character map.

```
char_map = {
    ' ': 0,'అ': 1,'ఆ': 2,'ఇ': 3,'ఈ': 4,'ఉ': 5,'ఊ': 6,'ఋ': 7,'ఎ': 8,'ఏ': 9,'ఐ': 10,
    'ఒ': 11,'ఓ': 12,'ఔ': 13,'క': 14,'ఖ': 15,'గ': 16,'ఘ': 17,'ఙ': 18,'చ': 19,'ఛ': 20,
    'జ': 21,'ఝ': 22,'ఞ': 23,'ట': 24,'ఠ': 25,'డ': 26,'ఢ': 27,'ణ': 28,'త': 29,'థ': 30,
    'ద': 31,'ధ': 32,'న': 33,'ప': 34,'ఫ': 35,'బ': 36,'భ': 37,'మ': 38,'య': 39,'ర': 40,
    'ఱ': 41,'ల': 42,'ళ': 43,'వ': 44,'శ': 45,'ష': 46,'స': 47,'హ': 48,'ా': 49,'ి': 50,'ీ': 51,
    'ు': 52,'ూ': 53,'ృ': 54,'ె': 55,'ే': 56,'ై': 57,'ొ': 58,'ో': 59,'ౌ': 60,'్': 61,
}
```

Figure 3.16: Character Map

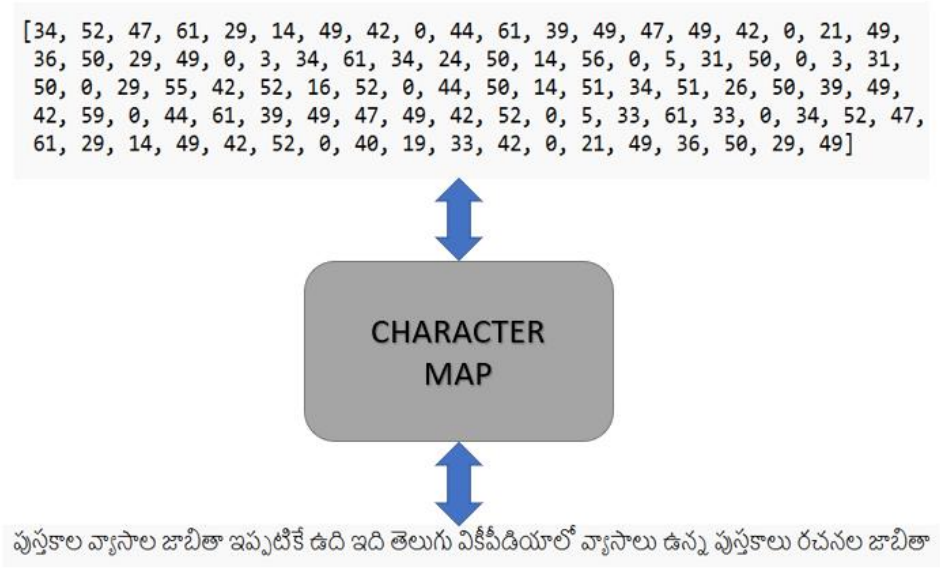


Figure 3.17: Block diagram of encode and decode unit

3.5 CTC LOSS AND CTC DECODE

CTC loss is a type of loss function used to train the neural network. Its goal is to align a sequence of inputs with a sequence of target outputs, where the length of the input sequence and the target sequence may differ. CTC achieves this by allowing the network to make "blank" predictions that do not correspond to any output, and by combining repeated output symbols into a single symbol. The CTC loss function computes the negative log-likelihood of the target sequence given the input sequence and the model parameters.

The CTC decoding algorithm works by collapsing the repeated symbols and removing the blank symbols from the network's output probabilities, and then finding the most likely sequence of symbols that corresponds to the remaining probabilities. This decoding process is done using an algorithm such as beam search or best path decoding.

Together, CTC loss and decode allow neural networks to effectively handle variable-length input and output sequences, noisy or partial annotations, and output sequences that are longer than the input sequence. This has led to significant improvements in many sequence prediction tasks, making CTC a popular choice for researchers and practitioners in the field [14].

3.6 DATA SET

Telugu ASR Training data set [11] is used for evaluating models on the Telugu ASR task. Since train/test splits were not provided for the Telugu data set, we propose our own evaluation procedure, splitting the data set into train and test sets in 9:1 ratio. Number of male speech files are 2154, Number of female speech files are 2294.

Table 3.1: Data Set Split

S.NO	SPLIT	NUMBER OF FILES
1	Train	4003
2	Test	445
3	Total	4448

CHAPTER 4

MODEL ARCHITECTURES

4.1. GENERAL

We have used two different model architectures for building low resource automatic speech recognition system, they are:

1) CNN architecture

2) CNN+LSTM architecture

4.2. CNN ARCHITECTURE

As in [10], CNN gives better results for speech recognition tasks .In this architecture we used two CNN layers the first layer consists of convolution layer with 64 filters followed by batch normalization followed by Relu activation and the second layer consists of convolution layer with 32 filters followed by batch normalization followed by Relu activation and the output of this layer will be reshaped and given to time distributed dense layer with SoftMax activation with 65(character map size) nodes .

```
Model: "TARUS"
_________________________________________________________________________________________________
 Layer (type)                                Output Shape                              Param #
=================================================================================================
 input (InputLayer)                          [(None, None, 128)]                       0

 expand_dim (Reshape)                        (None, None, 128, 1)                      0

 conv_1 (Conv2D)                             (None, None, 64, 32)                      288

 conv_1_bn (BatchNormalization)              (None, None, 64, 32)                      128

 conv_1_relu (ReLU)                          (None, None, 64, 32)                      0

 conv_2 (Conv2D)                             (None, None, 32, 32)                      9216

 conv_2_bn (BatchNormalization)              (None, None, 32, 32)                      128

 conv_2_relu (ReLU)                          (None, None, 32, 32)                      0

 reshape (Reshape)                           (None, None, 1024)                        0

 time_distributed (TimeDistributed)          (None, None, 65)                          66625

=================================================================================================
Total params: 76,385
Trainable params: 76,257
Non-trainable params: 128
_________________________________________________________________________________________________
```

Figure 4.1: CNN Model Summary

4.3. CNN+LSTM ARCHITECTURE

In [1] they used CNN+RNN architecture. As RNN has vanishing gradient problem here we are going to use LSTM. In this architecture we used two CNN layers the first layer consists of convolution layer with 64 filters followed by batch normalization followed by Relu activation and the second layer consists of convolution layer with 32 filters followed by batch normalization followed by Relu activation and the output of this layer will be reshaped and given LSTM layer with 64 units then to time distributed dense layer with SoftMax activation with 65(character map size) nodes[6].

```
Model: "TARUS"
_________________________________________________________________________________________________
 Layer (type)                                Output Shape                            Param #
=================================================================================================
 input (InputLayer)                          [(None, None, 128)]                     0

 expand_dim (Reshape)                        (None, None, 128, 1)                    0

 conv_1 (Conv2D)                             (None, None, 64, 32)                    288

 conv_1_bn (BatchNormalization)              (None, None, 64, 32)                    128

 conv_1_relu (ReLU)                          (None, None, 64, 32)                    0

 conv_2 (Conv2D)                             (None, None, 32, 32)                    9216

 conv_2_bn (BatchNormalization)              (None, None, 32, 32)                    128

 conv_2_relu (ReLU)                          (None, None, 32, 32)                    0

 reshape_3 (Reshape)                         (None, None, 1024)                      0

 lstm (LSTM)                                 (None, None, 64)                        278784

 time_distributed_3 (TimeDistributed)        (None, None, 65)                        4225

=================================================================================================
Total params: 292,769
Trainable params: 292,641
Non-trainable params: 128
```

Figure 4.2: CNN+LSTM Model Summary

4.4. TRANSFER LEARNING

Transfer learning by weights is a popular method of transfer learning where a pre-trained model's weights are used as a starting point for training a new model for a different but related task. This method involves using a pre-trained model that has already been trained on a large dataset as a starting point, and then reusing its learned weights to train a new model for a related task.

Transfer learning by weights can significantly reduce the time and resources required for training a new model from scratch. It also helps to improve the performance of the new model by leveraging the pre-trained model's learned representations, leading to faster convergence and better generalization on smaller datasets [6].

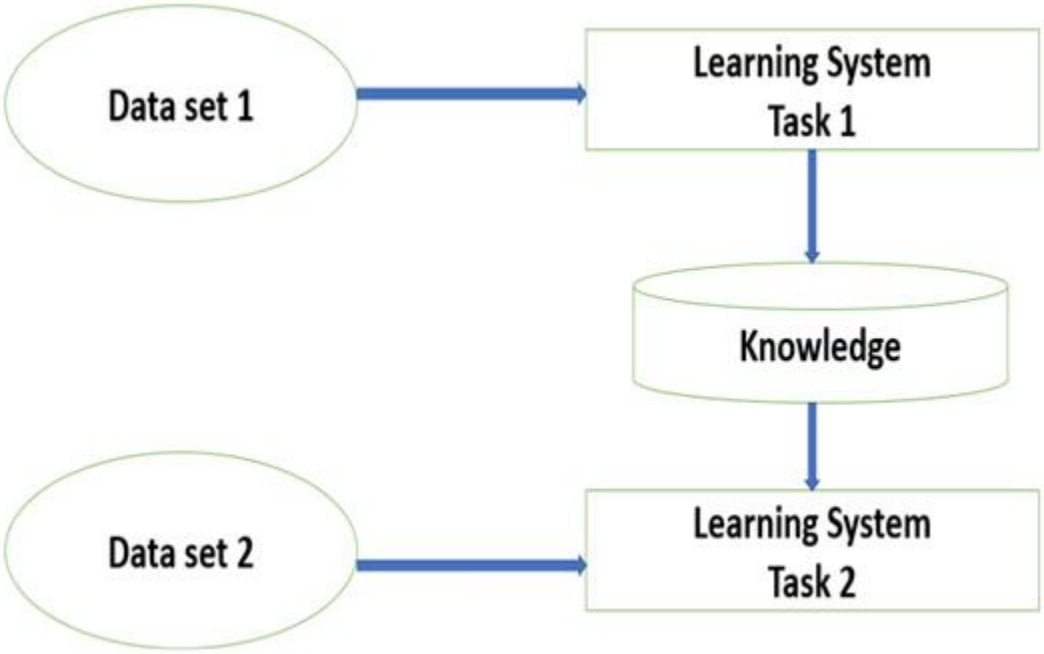


Figure 4.3: Block diagram of Transfer Learning

CASE 1:

Data set 1: English dataset

Data set 2: Telugu dataset

Learning System 1: CNN model

Learning System 2: CNN model

First the CNN model is trained with English data set and the weights are transferred to CNN model for Telugu data set.

CASE 2:

Data set 1: Telugu dataset

Data set 2: Telugu dataset

Learning System 1: CNN model

Learning System 2: CNN+LSTM model

First the CNN model is trained with Telugu data set and the weights are transferred to CNN+LSTM model for Telugu data set.

CHAPTER 5

RESULTS AND DISCUSSION

5.1. WORK DONE

In this phase of project, we developed a Low Resource Automatic Speech Recognition System using transfer learning technique.

We used Character Error Rate (CER) as the figure of merit, as the model deals with character recognition in each time frame.

CHARACTER ERROR RATE (CER): It represents the percentage of characters in the recognized text that differ from the original or expected text. In simple terms, CER is a measure of how many errors a text recognition system makes when transcribing text. The lower the CER, the higher the accuracy of the system.

5.2. RESULT

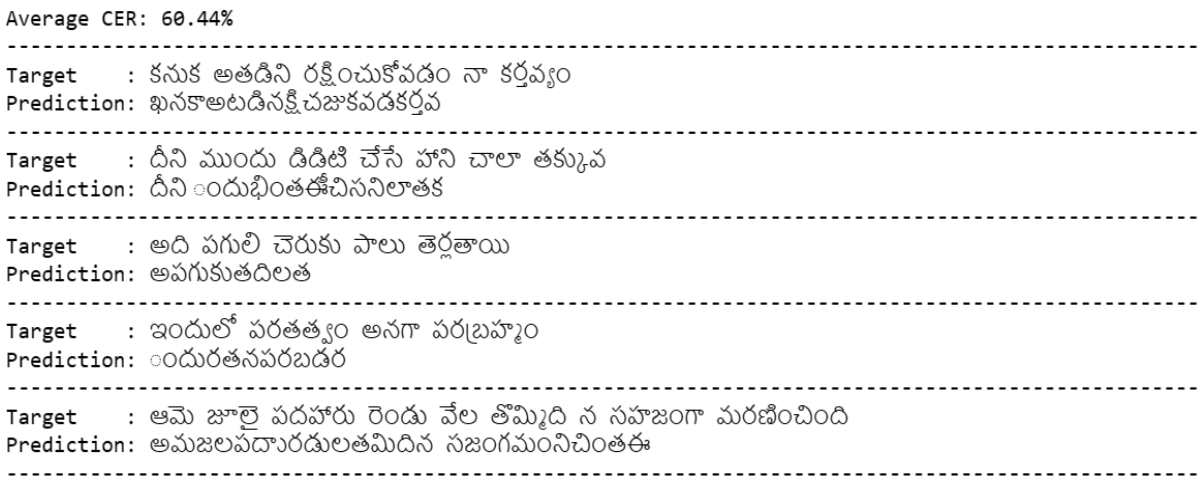


Figure 5.1: CNN Model Output

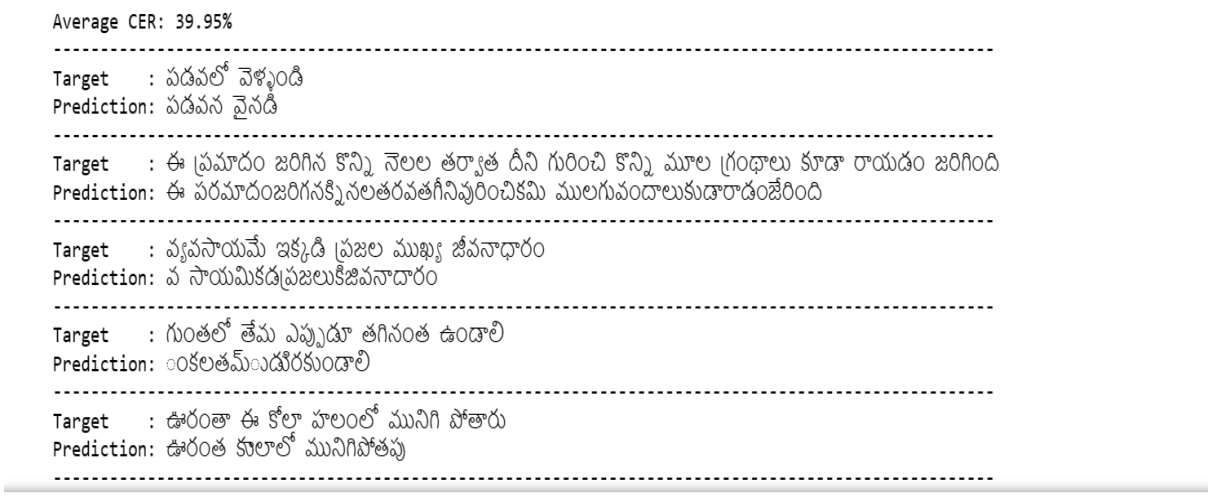


Figure 5.2: CNN+LSTM Model Output

Table 5.1: Final Results

S.NO	FEATURES	MODEL ARCHITECTURE	LOSS FUNCTION	AVERAGE CER
1	Mel-Spectrogram	CNN	CTC Loss	60.44%
2	Mel-Spectrogram	CNN+LSTM	CTC Loss	39.95%

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1. CONCLUSION

In this work we have developed a Low Resource Automatic Speech Recognition System comprising of CNN+LSTM based deep neural network trained using CTC loss function and transfer learning technique for Telugu language.

The model is developed to recognize the characters rather than the phonemes to which we need language model and pronunciation dictionary for correct sequence of sentence as in [2]. In this system we used character map as replacement of language model and pronunciation dictionary.

6.2. FUTURE SCOPE

- By changing the character map and the data set we can develop ASR system for any other language by changing the dense layer nodes to size of the character map for that language.
- By changing the model architectures like 2CNN+2LSTM,2CNN+3LSTM, CNN+4LSTM. The CER may decrease.
- As in [4] using Data Augmentation technique we can increase the data set and train the model to perform effectively in noisy environment.

REFERENCES

[1] Mandal, Sayan & Yadav, Sarthak & Rai, Atul. (2020). End-to-End Bengali Speech Recognition.

[2] M Ahmadi, N J Bailey, B S Hoyle. “Phoneme recognition using speech image (spectrogram)”. Published in IEEE: Proceedings of Third International Conference on Signal Processing (ICSP'96). doi: 10.1109/ICSIGP.1996.567353.

[3] Devaraj Adiga, Rishabh Kumar, Amrith Krishna, Preethi Jyothi, Ganesh Ramakrishnan, and Pawan Goyal, “Automatic speech recognition in Sanskrit: A new speech corpus and modeling insights,” in Proc. of the 59th Annual Meeting of the Association for Computational Linguistics (ACL Findings), 2021.

[4] B. Thai, R. Jimerson, D. Arcoraci, E. Prud'hommeaux and R. Ptucha, "Synthetic Data Augmentation for Improving Low-Resource ASR," 2019 IEEE Western New York Image and Signal Processing Workshop (WNYISPW), Rochester, NY, USA, 2019, pp. 1-9, doi: 10.1109/WNYIPW.2019.8923082.

[5] L. Besacier, E. Barnard, A. Karpov, and T. Schultz, “Automatic speech recognition for under-resourced languages: A survey,” Speech Communication, vol. 56, pp. 85 – 100, 2014.

[6] Bansal, Sameer and Kamper, Herman and Livescu, Karen and Lopez, Adam and Goldwater, Sharon. "Pre-training on high-resource speech recognition improves low-resource speech-to-text translation". Association for Computational Linguistics, doi:10.18653/v1/N19-1006, pp.58-68,2019

[7] V.-B. Le and L. Besacier, “Automatic speech recognition for under-resourced languages: Application to vietnamese language,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 17, no. 8, pp. 1471–1482, 2009.

[8] Yu Zhang, William Chan, and Navdeep Jaitly, “Very deep convolutional networks for end-to-end speech recognition,” in Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017, pp. 4845–484

[9] Afshine Amidi and Shervine Amidi, "VIP Cheatsheet: Convolutional Neural Networks”, November. 2018.

[10] Ethan Morris “Automatic Speech Recognition for Low-Resource and Morphologically Complex Languages,” Rochester Institute of Technology RIT Scholar Works ,2021.

[11] He, Fei and Chu, Shan-Hui Cathy and Kjartansson, Oddur and Rivera, Clara and Katanova, Anna and Gutkin, Alexander and Demirsahin, Isin and Johny, Cibu and Jansche, Martin and Sarin, Supheakmungkol and Pipatsrisawat, Knot. "Open-source Multi-speaker Speech Corpora for Building Gujarati, Kannada, Malayalam, Marathi, Tamil and Telugu Speech Synthesis Systems," in Proceedings of The 12th Language Resources and Evaluation Conference(LREC),may.2020,pp6494--6503,Available: https://www.aclweb.org/anthology/2020.lrec-1.800, ISBN:979-10-95546-34-4

[12] Musaev, Muhammadjon & Xujayorov, Ilyos & Ochilov, Mannon. (2019). Image Approach to Speech Recognition on CNN. 1-6. 10.1145/3386164.3389100.

[13] S. S. Holla, T. N. M. Kumar, J. R. Hiretanad, K. T. Deepak and A. V. Narasimhadhan, "End-to-End Speech Recognition for Low Resource Language Sanskrit using Self-Supervised Learning," 2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), Chennai, India, 2022, pp. 148-152, doi: 10.1109/WiSPNET54241.2022.9767118.

[14] K. -H. Lu and K. -Y. Chen, "A Context-Aware Knowledge Transferring Strategy for CTC-Based ASR," 2022 IEEE Spoken Language Technology Workshop (SLT), Doha, Qatar, 2023, pp. 60-67, doi: 10.1109/SLT54892.2023.10022825.