Proceedings of 10th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2021 Month: October Volume: 10

English Speech Recognition Using Convolution Neural Network, Gated Recurrent Unit and Connectionist Temporal Classification

Bishon Lamichhane [a], Bal Krishna Nyaupane [b], Smita Adhikari [c]

[a, b, c] *Department of Electronics and Computer Engineering, Pashchimanchal Campus, IOE, Tribhuvan University, Nepal*
Corresponding Email: [a] lcbishon@gmail.com, [b] bkn@wrc.edu.np, [c] adsmita1@gmail.com

Abstract
Conversion of an audio into equivalent text is called speech recognition system. The preferred way of communication between people is Speech. Making machines capable of conversing with humans is a difficult but necessary task. A vital first step is for machines to be able to be able to hear and understand what humans are saying. As a result, voice recognition becomes a critical tool for bridging the gap between robots and people. This work implements CNN(Convolution Neural Network),GRU(Gated Recurrent Unit) and CTC(Connectionist Temporal Classification) to convert English audio into equivalent English text. A deep neural network solves two closely related tasks. It learns to recognize phonemes and to formulate grammar rules at the same time. In fact, a model is able to parallel and accurate build both of them, when a training corpus is large enough. CNN is used for extracting time and frequency features. GRU is used for processing sequential data. CTC allows GRU to learn over train data. After processing data through CNN,GRU and CTC English Text is obtained as output.

Keywords
Speech Recognition,Artificial Neural Network, CNN, GRU, CTC

1. Introduction

Speech is the most common manner of human beings expressing their thoughts through speech. Speech has been in practice since the human civilization in different regional languages. Speech recognition is the process of using computers or programs to translate speech into equivalent textual representation. With the availability of higher computing power and higher collection of data, speech recognition is going to make computers easier to understand human speech. Bell Lab in 1952 created the first speech recognition system, a digit recognizer[1]. Since then, there has been a lot of research on voice recognition, and it has gotten a lot of attention.

The automatic speech processing began in the 1930s, although the first more widely used ASR systems appeared in the '70s [2, 3]. Initially, speech recognition systems were limited to recognizing more than a few hundred words, and often required the breaks between words spoken in sequence. The advancement of computers in the '90s has resulted in several complex ASR systems based on Hidden Markov Models (HMM) [3]. In those days speech recognition systems were composed of acoustic models, pronunciation models and language models. Each component was separately designed and fine tuned on an independent dataset, so in consequence, the entire process of building such a system was time-consuming and required expertise. In those days, speech recognition systems were composed of many stages, including handcrafted process of features extraction, and mentioned models HMM. Each of the components required fine-tuning, so in consequence, the entire process of building such system is time-consuming and required expertise. These systems are inflexible, so any changes would require long re-adjustments of the entire system.

Despite the notable improvements the ASR systems could not compete with the human quality of speech recognition. The situation is changed when an automatic speech recognition system fully based on the Recurrent Neural Network (RNN) was introduced in 2014 [4].Unlike traditional systems based on HMM, the entire process of automatic speech recognition is accomplished end-to-end by a single deep neural network.The key idea is to use the Connectionist Temporal Classification (CTC) algorithm [5], which

enables either to train a model or to do inference.

In the field of speech recognition, significant progress has been made during the recent decades.The use of significantly more training data is now possible because to advances in computer resources as well as parallel methods. Speech recognition algorithms may now be trained lots of audio data. Deep learning has recently piqued researchers' interest due to its ability to increase performance across a broad variety of tasks. in conjunction with the advancement of speech recognition technology, numerous businesses have included speech recognition into their products, such as Google's watch,Apple's Siri and Microsoft's Skype speech translation. Voice recognition technology is undeniably entering our daily lives and, eventually, transforming our lifestyles.

English Speech Recognition is a sequence to sequence problem where input has variable length of audio whereas output has equivalent representation of the text but which is also not fixed in length. In order to train network with input speech we cannot consider whole signal as audio signal is continuous series in time domain. In order to process the signal for machine learning it has to be sampled and converted into discrete series without losing any features. Traditionally spectrogram was directly fed into the RNN which increased the number of input features. The increased features increased CPU utilization and made the model learn in time consuming manner. When spectrogram is passed into the CNN then the number of input features for spectrogram decreases and more meaningful features can be extracted from the spectrogram.

2. Literature Review

An advanced artificial neural network (ANN) with directed memory cycles is called a recurrent neural network (RNN).Recurrent neural networks can build on previous types of networks that have fixed-size input and output vectors.

2.1 Related Work

Abdel-Hamid, O. et.al. (2014) [6] suggested a method for implementing CNN in speech recognition, proposing a constrained weight sharing strategy that can better handle speech elements. They show in this research that utilizing convolutional neural networks can reduce error rates even more. The paper begins with a brief overview of the fundamental CNN and how it can be applied to speech recognition. They also offer a limited-weight-sharing technique for enhanced speech modeling. CNNs' unique structure, which includes local connection, weight sharing, and pooling, demonstrates some invariance to tiny shifts in speech parameters along the frequency axis, which is crucial when dealing with speaker and environment variances.On the TIMIT large vocabulary speech recognition and phone recognition tasks, experimental data demonstrate that CNNs lower mistake rates by 6% to 10% when compared to DNNs.Values can be regarded as 2-D feature maps when CNN is employed for image processing. When CNN is employed for voice recognition, however, the input "picture" is treated as a "spectrogram" of speech.

Sak, H. et.al. (2014, September) [7] suggested that for large-scale Acoustic Modeling in voice recognition, the LSTM RNN architecture is effective. The researchers discovered that some jobs necessitate a large number of input, output, and memory cells in order to store temporal contextual data. As a result, LSTM model learning will be computationally expensive. They also suggested using many LSTM layers in a deep LSTMP. To compute the parameter gradients on short subsequences of training utterances, each layer contains distinct stacked recurrent projection layers. The BPTT (truncated backpropagation through time) algorithm was utilized. Asynchronous Stochastic Gradient Descent (AGSD) is used to optimize network parameters.

According to Ilya Sutskever et al. (2014) [8], the LSTM technique will likely perform well on difficult sequence to sequence challenges. DNNs (Deep Neural Networks) are advanced learning models that have excelled in difficult learning challenges. DNNs can be used to map sequences to sequences when large labeled training sets are available; but, when big labeled training sets are not available, DNNs cannot be used to map sequences to sequences. In this paper, they present an universal end-to-end approach to sequence learning, based on only a few assumptions regarding sequence structure. Their approach employs a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a fixed-dimensional vector,the target sequence from the vector is then decoded using a deep LSTM.Researchers' main conclusion is that the LSTM's translations attain a BLEU score of 34.8 on the whole test set on an English to French translation assignment from the WMT'14 dataset, with the LSTM's BLEU score

penalized for out-of-vocabulary words.Furthermore, the LSTM has no trouble with extended sentences. On the same dataset, a phrase-based SMT system receives a BLEU score of 33.3.The BLEU score climbed to 36.5 when they used the LSTM to rerank the 1000 hypotheses generated by the SMT system, which is close to the previous best performance on this test.They discovered that reversing the order of words in all source sentences (but not in target sentences) considerably enhanced the LSTM's performance, owing to the creation of multiple short-term dependencies between the source and target phrases, which simplified the optimization problem.

M. Ravanelli et al. (2018 April) [9] proposed a streamlined architecture for gated recurrent units (GRUs), one of the most popular RNN models.This work makes a two-fold contribution: They begin by examining the purpose of the reset gate, demonstrating that there is significant redundancy with the update gate. As a result, they recommend removing the former from the GRU architecture, resulting in a single-gate model that is more efficient and compact. Second, they recommend that hyperbolic tangent activations be replaced with rectified linear unit activations. This version works well with batch normalization, and it may be able to assist the model understand long-term dependencies without causing numerical problems. The proposed architecture, dubbed light GRU, not only cuts per-epoch training time by more than 30% compared to a standard GRU, but also improves recognition accuracy consistently across different tasks, input features, noisy conditions, and ASR paradigms, ranging from standard DNN-HMM speech recognizers to end-to-end connectionist temporal cassification models.The suggested Li-GRU architecture is a reduced variant of a normal GRU that considers ReLU activations instead of the reset gate. Batch normalization is also utilized to increase the system's performance and to reduce numerical instabilities caused by ReLU non-linearities. Experiments on various ASR paradigms, tasks, features, and environmental variables have proven that the suggested model is effective. In fact, the Li-GRU not only improves recognition performance but also reduces computational complexity, saving more than 30% of training time when compared to a regular GRU. Future efforts will be focused on expanding this work to other speech-based tasks, such as speech enhancement and speech separation, as well as looking into Li-application GRUs in other domains.

FAN et al.(2021) [10] proposed a gated recurrent fusion (GRF) technique with joint training architecture for robust end-to-end ASR. The noisy and improved features are dynamically combined using the GRF method. As a result, the GRF can not only eliminate noise signals from improved features, but to eliminate speech distortion, it can also learn the raw fine structures from noisy characteristics. Speech augmentation, GRF, and speech recognition are all part of the suggested method. To begin, the input speech is enhanced using a mask-based speech improvement network. Second, the GRF is used to address the issue of speech distortion. Third, to improve ASR performance, the state-of-the-art speech transformer algorithm is used as the speech recognition component.Finally, the combined training framework is used to simultaneously optimize these three components. Their research is based on the AISHELL-1 open-source Mandarin voice corpora. The proposed method reduces the relative character error rate (CER) by 10.04% when compared to the standard joint enhancement and transformer method that solely uses enhanced features, according to experimental results. Their proposed strategy can provide improved performance, especially for low signal-to-noise ratios.

They suggest a hybrid enhancement and speech transformer training technique for robust end-to-end speech recognition in this research, using gated recurrent fusion. The joint training compositional technique is utilized to optimize both enhancement and speech recognition at the same time. They also use the gated recurrent fusion algorithm to fuse the noisy and upgraded features in order to address the voice distortion problem and extract more robust features for end-to-end ASR. Experiments on Mandarin AISHELL-1[10] show that their proposed solution is effective for end-to-end ASR and can effectively handle the speech distortion problem.

3. Methodology

Speech recognition system works in stages. Speech is first collected, pre-processed and then it follows training and testing phase. Block diagram of model is shown in Figure 1.

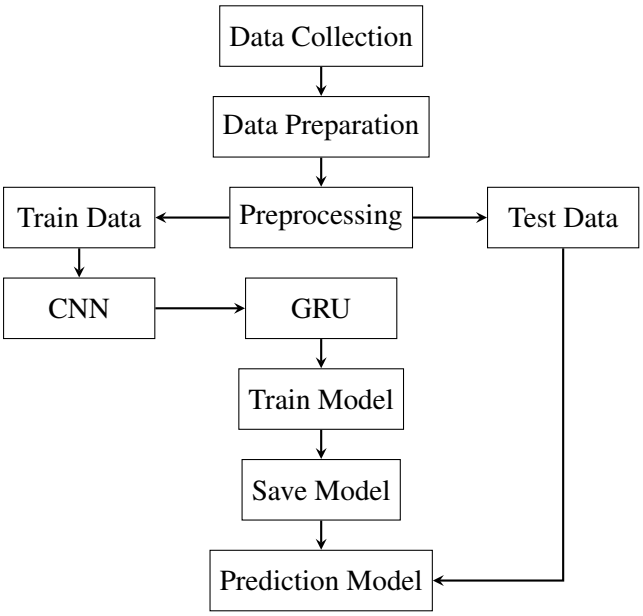


Figure 1: Block Diagram Of The Model

3.1 Dataset

Large amount of data is required to train GRU, so it is not feasible to collect data manually.LibriSpeech Dataset[11] is used in this work. LibriSpeech is a corpus of read speech, based on LibriVox's public domain audio books. The dataset used in this thesis work is in English language.

3.2 Pre-Processing of audio signals

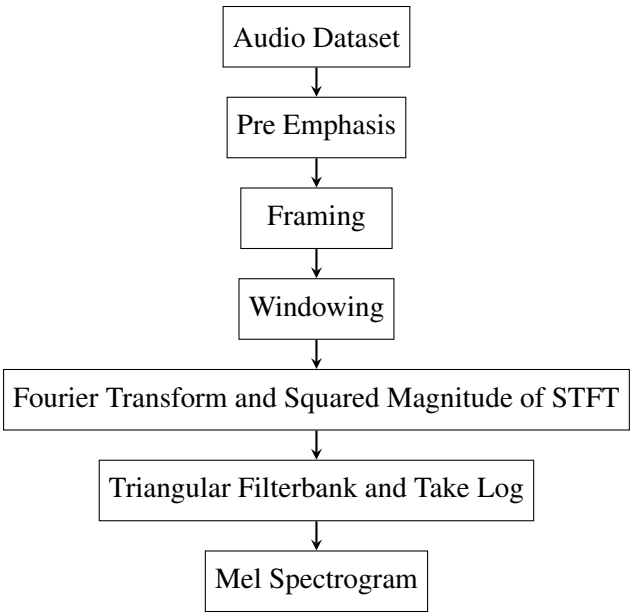


Figure 2: Block Diagram Of Pre Processing

The training phase is marked by the extraction of features from a large number of speech instances ("training data"), whereas the testing phase is marked by the extraction of features from "testing data," and the validation data is marked by the extraction of features from "validation data." One of the sample named "8461-281231-0038" is taken from training dataset whose waveform when plotted looks as shown in the Figure 3.

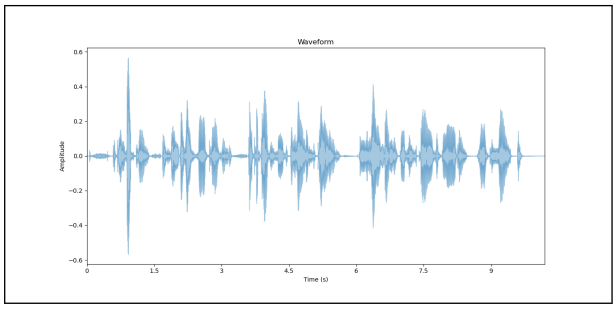


Figure 3: Waveform of Training Sample

3.2.1 Pre-Emphasis

The raw wave is pre emphasized to amplify higher frequencies.

$$y[n] = x[n] - \alpha.x[n-1] \tag{1}$$

In this thesis the value of α used is 0.97. After the pre-emphasis the waveform looks like shown below.

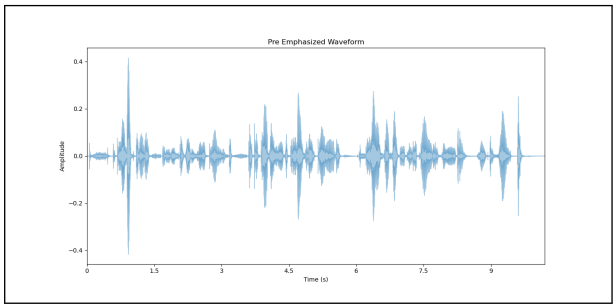


Figure 4: Waveform of Training Sample after Pre-Emphasis

3.2.2 Framing

After pre emphasis the audio is splited into small frames.In voice processing, Frame sizes typically range from 20 to 40 milliseconds, with a 50% (+/-10%) gap between successive frames. Settings used in this work are 20 ms for the frame size, and a 10 ms stride.

3.2.3 Window

Following the signal's splitting into a number of frames, a Hanning window of given form is applied:

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

where, $0 \leq n \leq N-1$ N is window length. Given the speech signal x[n], the short time signal $x_m[n]$ of the frame m is calculated as:

$$x_m[n] = x[n].w_m[n] \quad (3)$$

with $w_m[n]$ is window function. Window function that is applied in this thesis work is hanning window

3.3 Fourier-Transform and Power Spectrum

Each frame m of a signal is Fourier transformed, and the results are collected in the matrix, which gathers the magnitude for each point in time and frequency. Which is expressed as:

$$X(m,w) = \sum_{n=-\infty}^{\infty} x[m] * w[n-m]xe^{-jt} \quad (4)$$

After framing and windowing Short Time Fourier Transform is applied to each frame using the following equation.

$$P = \frac{|FFT(x_i)|^2}{N} \quad (5)$$

The hop length used for this thesis work is 160 and N used is 512.

3.3.1 Filter Banks

After Fourier-Transform filter banks are applied in order to obatain mel scale from hertz.

$$m = 2595\log_{10}(1+\frac{f}{700}) \quad (6)$$

$$f = 700(10^{m/2595}-1) \quad (7)$$

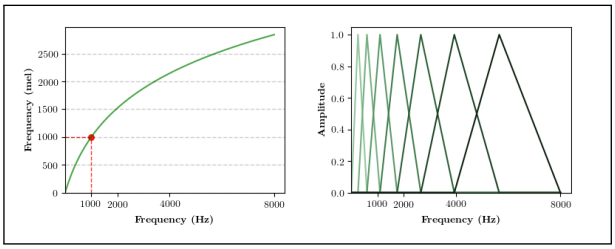


Figure 5: The relationship between the frequency scale and mel scale is shown on the left. The Mel-scale filter-bank composed of 7 filters is presented on the right

The following equation can be used to model this.

$$H_m(k) = \begin{cases} 0 & K < f(m-1) \\ \frac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1) \leq k \leq f(m) \\ \frac{f(m+1)-k}{f(m+1)-f(m)} & f(m) \leq k \leq f(m+1) \\ 0 & k > f(m+1) \end{cases} \quad (8)$$

Where, H_m denotes magnitude of filter m, the frequency is denoted by k and the vector f contains m+ 2 linearly spaced filter border values. In the processing of computing mel spectrogram, the hop length of 0.01 ms(160) is used with sampling rate of 16 KHz and 160 mel filters are selected. Thus, the mel spectrogram with n frames and 160 mel frequency bins is calculated. The mel spectrogram can be considered as a grid or a matrix. The input of the CNN network is the 160 vector for each time frame of 0.02 ms(320) in this research.

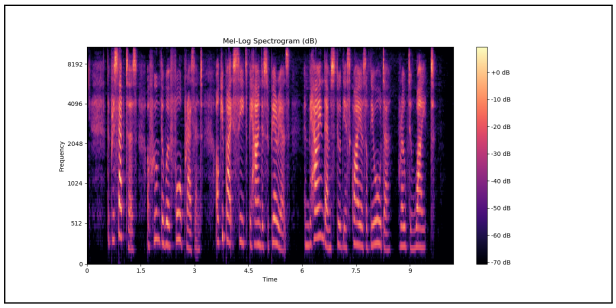


Figure 6: Mel Spectrogram of Training Sample

3.4 Convolution Neural Network

After the generation of Mel-log spectrogram, the Mel-log spectrogram is fed into Convolution Neural Network with two-dimensional convolution layer. In the two-dimensional convolutional layer shown here first dimension is time and second dimension is frequency. After the application of convolution, the length of the sequence to be processed by subsequent recurrent layers is halved, which reduces the duration of experiments almost two times.

The first layer after the feature extraction process of the Model is a two-dimensional (time-and-frequency domain) convolutional layer. The layer analyses a small context bounded by a receptive field with size in time cf and frequency cf.The layer representation is defined by h_l, assuming that h_0 represents input data x. Convolutional layer activations can be written for the i-th activation at time t and frequency f as:

$$h^l{}_{t,f,i} = \mathscr{F}(w^l{}_i * h^{l-1}{}_{t-ct,t+ct:f-cf,f+cf}) \quad (9)$$

where * means a convolve of an i-th filter with receptive field of input data, and $\mathscr{F}(.)$ is an unary function.The selected function in this work is Clipped Rectified Linear (ReLU).

$$\sigma(x) = min\{max\{x,0\},20\} \tag{10}$$

In this thesis work the input to the CNN is a matrix of size no_of_frames*160 and 2 layers of CNN are used with 32 as the number of filters in both layer, kernel size of [11, 41] for first layer and [11, 21] for the second layer and strides of [2,2] is used in first layer and [1, 2] is used in second layer. Before passing to the second layer, the output of the first layer is batch normalized and sent via the reLu activation function. Similarly, before being fed into GRU, reLu activation function is used to batch normalize the output of second layer.CNN is used in this thesis work with the aim of extracting more detail feature in an efficient manner.

3.5 Network Training with GRU

Unlike standard feed forward network GRU use past output as feedback so, the past output also contributes in current output. Having capacity to remember GRU are suitable for sequential and time dependent data such as speech. The output from the CNN consists of input features for training samples. The input features are then fed into GRU for training. The network consists of 5 layers of GRU consisting of 800 neurons each. In this thesis work the network consists of 5 layers of GRU.

3.6 Connectionist Temporal Classification as cost function

The CTC Loss Function $\mathscr{L}$ for a single (X,Y) pair is defined as:

$$\mathbb{P}(Y/X) = \sum_{A \varepsilon A_{X,Y}} \prod_{t=1}^{T} p_t(a_t/X) \tag{11}$$

$$\mathscr{L}_{CTC} = -log\mathbb{P}(Y/X) \tag{12}$$

where $A_{X,Y}$ describes valid alignments. The parameters of the model are chosen to minimize the negative log-likelihood. The CTC conditional probability marginalizes over all valid alignments, computing the probability for each one individually. The straightforward approach scoring each alignment and sum them up fails due to the massive number of alignments. We can quickly compute the loss function using the dynamic programming approach.. The crucial insight is that two alignments can be merged if they have reached the same output at the same time. The CTC Loss is fully differentiable with respect to prediction at each time step, since the objective just sums the product of them. In result, it is possible to compute the gradient, and run the backpropagation as usual. The output from the GRU is passed through CTC loss function. The function calculates the error and backpropagates error to previous layer for parameter tunning.

3.7 Connectionist Temporal Classification as decoder

The cost function Connectionist Temporal Classification (CTC) is used to train Recurrent Neural Networks (RNNs) to label unsegmented input sequence data.Real world data for sequence learning requires prediction from noisy and unsegmented data. Let's consider a scenario where input sequence X=[x1 ,x2,. . . ,xT] (speech signal) is to be mapped to output sequences Y=[y1,y2,. . . ,yU] (transcript) . There are some problems

- The lengths of XX and YY can differ.
- The lengths of XX and YY can be in different proportions.
- There is no accurate alignment (correspondence of the elements) of XX and YY.

The CTC converts the network outputs into a conditional probability distribution across label sequences.

3.8 Output Decoding

It is the final layer of network. The network output vector is normalized using SoftMax function. The number of outputs from SoftMax layer is equal to count of all characters in language domain (in English language all alphabet) plus space, blank symbol and '. Since output is normalized so the sum of the output is equal to one. Output vector from SoftMax layer is associated with our character set.

3.8.1 Max Decoding

In max decoding vector element having highest probability is picked and its equivalent character is

chosen. It is predicated on the idea that the most likely path will correlate to the most likely labeling.But this approach has a problem let's say if a wrong character is chosen at start point then whole result is affected by this decision.

4. Epilogue

4.1 Data Analysis

For training and testing,LibriSpeech dataset prepared by Vassil Panayotov with the help of Daniel Povey is used. LibriSpeech is a corpus of read speech, based on LibriVox's public domain audio books. The dataset used in this thesis work is in English language. The LibriSpeech corpus is made up of 1000 hours of speech sampled at 44.1kHz and down sampled to 16kHz derived from the LibriVox project's audiobooks.The minimum duration of the audio is 4 seconds and maximum duration of audio clips is 14 seconds.

4.2 Experimental Results and Discussion

Different CSV files are created for training, validation and testing The CSV files contain the audio filename and equivalent textual representation of the given audio. Different experiments are conducted in this research. The model built is trained and experimented on different amount of dataset.The model is trained with 12000 dataset.The results obtained are explained are given in table 1.Greedy decoder is used for decoding.Automatic speech recognition systems are evaluated on the grounds of Word Error Rate(WER).

The Word Error Rate is calculated using the equation:

$$WER = \frac{S+I+D}{N} \tag{13}$$

where,
S is Substitutions count,
D is Deletions count,
I is Insertions count and
N is Words count in reference
If the correct text is as given below:
this is a libravox recording all libravox recordings are in the public domain for more information or to volunteer please a visit libravox dot org

and here's how that sentence is translated with model: this is a libera ox recording all librvox recordings are in the public domain for more information nor to volunteer please a viset liber of ox dot org Here,
Count of substitutions(S)=5
Count of Deletions(D)=0
Count of Insertions(I)=3
Count of words in reference(N)=25
WER=(5+0+3)/25=8/25=0.32
The character error rate is defined in similar way as:

$$CER = \frac{S+I+D}{N} \tag{14}$$

where
S is Substitutions count,
D is Deletions count,
I is Insertions count and
N is characters count in reference
If libravox is translated as libera ox The CER for the given text is calculated as:
Count of substitutions(S)=1
Count of Deletions(D)=1
Count of Insertions(I)=1
Count of characters in reference(N)=8
CER=(1+1+1)/8=3/8=0.375

4.3 Training, testing model and checking the performance

For training CNN is followed by multiple layers of GRU. The error is calculated using CTC loss function and backpropagated during the training.

Experiment The model is trained with 12000 and the train to test ratio used was 60 for training, 20 for testing and 20 for validation in batch size 32,the result obtained is as below in table

Table 1: WER and CER from decoder

Decoder	WER	CER
Greedy Decoder	20.92	13.77

Ground Truth Vs Prediction The test sample are passed through the trained model.Test samples pass through different steps before the final prediction is obtained.The features of test samples are first extracted and then it is feed into the trained model where it passes through CNN and RNN followed by decoder. In the following section a sample from prediction data has been considered.

Sample: this is a libravox recording all libravox recordings are in the public domain for more information or to volunteer please a visit libravox dot org

Which is predicted in the following manner.

First the test audio in raw form can be visualized as:

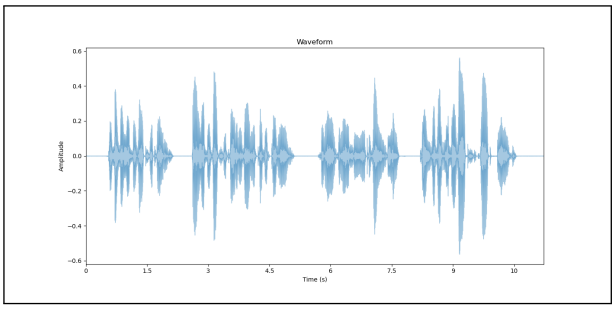


Figure 7: Waveform of Sample1

The raw wave is then pre emphasized:

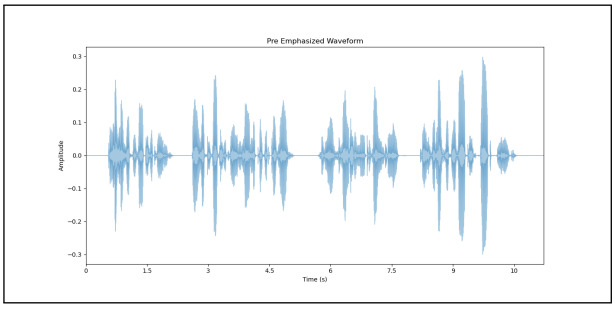


Figure 8: Pre-Emphasized Waveform of Sample1

After pre emphasis, framing,windowing is done to calculate the spectrogram of the audio signal which is converted into mel scale to obtain mel spectrogram as:

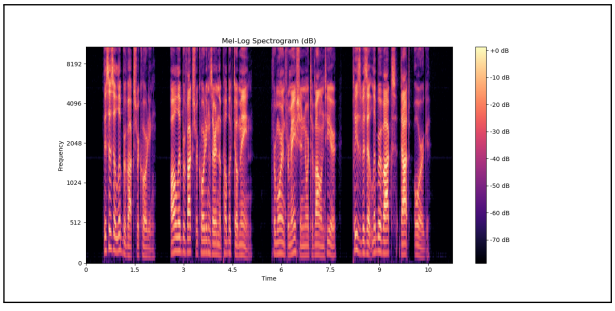
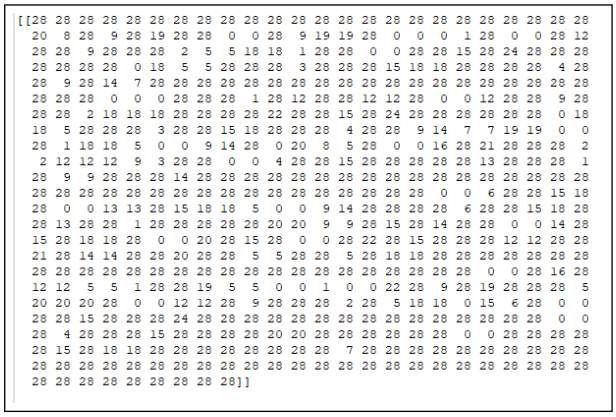


Figure 9: Mel Spectogram of Sample1

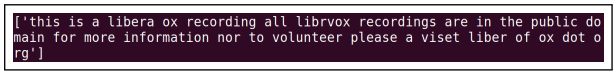
Spectograms are passed to the network and the probability of character is obtained.From the probability decoder chooses best suitable candidates.



```
[[28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28
  20  8 28  9 28 19 28 28  0  0 28  9 19 19 28  0  0  0  1 28  0  0 28 12
  28 28  9 28 28 28  2  5  5 18 18  1 28 28  0  0 28 28 15 28 24 28 28 28
  28 28 28 28  0 18  5  5 28 28 28  3 28 28 28 15 18 18 28 28 28 28  4 28
  28  9 28 14  7 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28
  28 28 28  0  0  0 28 28 28  1 28 12 28 28 12 12 28  0  0 12 28 28  9 28
  28 28  2 18 18 18 28 28 28 22 28 28 15 28 24 28 28 28 28 28 28  0 18
  18  5 28 28 28  3 28 28 15 18 28 28 28  4 28 28  9 14  7  7 19 19  0  0
  28  1 18 18  5  0  0  9 14 28  0 20  8  5 28  0  0 16 28 21 28 28 28  2
   2 12 12 12  9  3 28 28  0  0  4 28 28 15 28 28 28 28 28 13 28 28 28  1
  28  9  9 28 28 28 14 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28
  28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28  0  0  6 28 28 15 18
  28  0  0 13 13 28 15 18 18  5  0  0  9 14 28 28 28 28  6 28 28 15 18 28
  28 13 28 28  1 28 28 28 28 28 20 20  9  9 28 15 28 14 28 28  0  0 14 28
  15 28 18 18 28  0  0 20 28 15 28  0  0 28 22 28 15 28 28 28 12 12 28 28
  21 28 14 14 28 28 20 28 28  5  5 28 28  5 28 18 18 28 28 28 28 28 28 28
  28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28  0  0 28 16 28
  12 12  5  5  1 28 28 19  5  5  0  0  1  0  0 22 28  9 28 19 28 28 28  5
  20 20 20 28  0  0 12 12 28  9 28 28 28  2 28  5 18 18  0 15  6 28  0  0
  28 28 15 28 28 28 24 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28  0  0
  28  4 28 28 28 15 28 28 28 28 20 20 28 28 28 28 28 28 28  0  0 28 28 28 28
  28 15 28 18 18 28 28 28 28 28 28 28  7 28 28 28 28 28 28 28 28 28 28 28
  28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28 28
  28 28 28 28 28 28 28 28 28]]
```

Figure 10: Probability Distribution Over Different Class

The repeated class are removed.The numerical value of the class is replaced by alphabetical values and the final result is obtained. Class 0 represents white space 1 represents a and so on. The result obtained is shown below:



```
['this is a libera ox recording all librvox recordings are in the public do
main for more information nor to volunteer please a viset liber of ox dot o
rg']
```

Figure 11: Prediction of the Input

Ground Truth Vs Model Prediction

Ground Truth:
this is a libravox recording all libravox recordings are in the public domain for more information or to volunteer please a visit libravox dot org

Model Prediction:
this is a libera ox recording all librvox recordings are in the public domain for more information nor to volunteer please a viset liber of ox dot org

5. Discussion and Conclusion

The model is tested on a small collection of data (12000).The accuracy of GRU across a given dataset is determined by many factors like learning rate and the number of epochs.The algorithm's accuracy is influenced by a number of parameters, including the quality of the training dataset, the testing environment, the quantity of datasets, and the algorithm's complexity and so forth.If we can collect a large amount of data, the accuracy of the model can be improved, and it can be trained in different situations to produce varied results. Here the work is limited

only to 12000 data, but it can be extended to higher number with higher computing power.

The contribution to the WER and CER are caused by many factors.Some mistakes are caused by the space, which has a high impact on the WER score. In fact, only a single misspelled space in the sequence composed of four words brings up to 50% of the WER score but CER doesn't increase significantly. The largest group of misspellings is caused by the wrong character prediction, which represents the substitution operation. If a single character is predicted wrong then it contributes more to the WER. The letters which sound similar, but depend on the context are written differently, such as t and d, p and b and b and v which contributes greatly to mistake character prediction.

We may deduce from the completed work that the GRU can be used in situations where the current output is dependent on previous states, such as voice recognition. The speech recognition systems can be implemented in the areas where voice commands are common such as assisting the blinds. Similarly, speech recognition increased the interest of people in regional language which helps to prevents languages from being extinct. Speech recognition systems are affected by the number of factors like noise, quality of recorder, accents of the speakers and so on. Although real time applications of Speech Recognition systems are in noisy environments they can't perform well in such situations.

6. Future Enhancements

The developed model is not limited only to english language but can be implemented over other language without having bother about the input and output alignment.The only thing that needs to be focused while implementing to new language is the number of labels needed for the particular language.If we want to implement this model to the Nepali speech then 29 labels won't be sufficient, we need to consider all the alphabets available in Nepali language.

The model can be implemented with large amount of data. If huge amount of data and resource is available then the WER and CER can be further minimized. The accuracy of the system can be further improved if the training dataset contains noisy audio.If the audio is noisy then pre processing involves more steps in order to obtain desired output.

References

[1] Roberto Pieraccini and ICSI Director. From audrey to siri. *Is speech recognition a solved problem*, 23, 2012.

[2] B. T. Lowerre. *The Harpy speech recognition system*. PhD thesis, Carnegie-Mellon Univ., Pittsburgh, PA., April 1976.

[3] B. Juang and Lawrence Rabiner. Automatic speech recognition - a brief history of the technology development. 01 2005.

[4] A. Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, 2014.

[5] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural 'networks. volume 2006, pages 369–376, 01 2006.

[6] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10):1533–1545, 2014.

[7] Hasim Sak, Andrew W. Senior, and Françoise Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH*, pages 338–342, 2014.

[8] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.

[9] Mirco Ravanelli, Philemon Brakel, Maurizio Omologo, and Yoshua Bengio. Light gated recurrent units for speech recognition. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(2):92–102, 2018.

[10] Cunhang Fan, Jiangyan Yi, Jianhua Tao, Zhengkun Tian, Bin Liu, and Zhengqi Wen. Gated recurrent fusion with joint training framework for robust end-to-end speech recognition, 2020.

[11] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.