International Journal of Innovative
Computing, Information and Control
Volume **19**, Number **6**, December **2023**

ICIC International ©2023 ISSN 1349-4198
pp. **1769–1784**

NEPALI SPEECH RECOGNITION USING SELF-ATTENTION NETWORKS

Basanta Joshi and Rupesh Shrestha*

Department of Electronics and Computer Engineering, Pulchowk Campus
Institute of Engineering
Tribhuvan University
Lalitpur 44700, Nepal
basanta@ioe.edu.np; *Corresponding author: 075msice018.rupesh@pcampus.edu.np

Received February 2023; revised June 2023

Abstract. *Automatic speech recognition system is able to transcribe an audio speech input to text. Attention-based Transformer networks are used to train over two Nepali Speech corpus, in which one is speech related to general subject matters while the other is based on Nepali Sports News context. Short-Time Fourier Transform (STFT) has been used to convert audio speech to the frequency domain and Convolutional Neural Network (CNN) is used to extract features from the resulting spectrogram. These audio features are passed to Transformer along with corresponding Nepali characters during the training. The sequence of Nepali characters is generated as output when the output of the Transformer is passed to the softmax layer with Connectionist Temporal Classification (CTC) decoding. A Character Error Rate (CER) of 55.65% was observed for speech corpus with mixed contexts while a character error rate of 19.55% was observed for Nepali Speech recognition for speech corpus based on Nepali Sports News context. The number of encoder-decoder layers and batch size has been optimized and tuned to improve the performance of Nepali Automatic Speech Recognition (ASR) using attention-based Transformer networks on the OpenSLR dataset to a CER of 19.47% and Word Error Rate (WER) of 38.69%.*

Keywords: Automatic speech recognition, Transformer, Attention, Nepali Speech recognition

1. **Introduction.** Speech has been realized as an effective medium in the task of expressing human feelings and thoughts. Therefore, speech recognition has been implemented in modern machines. This has been achieved by training them to recognize different speech components of human speech. The first step is to extract useful features from the raw speech. The second step involves generating an acoustic model where speech is represented as a combination of different phones.

ASR's objective is to convert speech into its textual representation. Machine learning and artificial intelligence have assisted in the development of speech recognition. Development of ASR is assisting human-computer interaction and interaction of disabled; ASR has helped persons with dysarthria speech to communicate efficiently which has been a great motivation for ASR-related research [1]. As the English language has been used as the standard language in the computer in the majority of parts of the world, the development of speech recognition technology has been focused on the English language and there has been significant improvement in speech recognition in the English language.

Although there are so many benefits of Nepali ASR, it is in its infant stage [2]. The major contribution of this work is the development of a Transformer-based model using

DOI: 10.24507/ijicic.19.06.1769

raw audio features using STFT of the audio and then visual features are extracted using CNN technology.

This paper is organized into five sections. The current section (Introduction) introduces the problem context and highlights the contribution of the work. The Literature Review discusses the related theory and work that has been carried out on the subject. The Methodology section describes the way the research work is conducted such as audio feature extraction, training of ASR, and approach for validation of the developed model. The results of the training, validation of the developed Transformer-based model, and the observations are discussed in the Results and Analysis Section. Finally, the Conclusions section summarizes the work and also highlights the limitations of this work.

2. **Literature Review.** Speech recognition nowadays is the result of the efforts and contributions of researchers over many decades. A single-speaker isolated digit recognition system using format frequencies which are estimated in each digit's vowel regions in 1952 [3]. In 1970's, Furui pioneered the use of a composite of instantaneous cepstral coefficients as essential spectral characteristics for speech recognition [4]. Another major achievement in speech recognition was the use of the Hidden Markov Model (HMM) in 1989 [5].

Later IBM emphasized on creation of a language model (grammar) structure, which was defined by statistical syntactical rules specifying how likely a succession of symbols inside a language (e.g., phonemes or words) will occur in the signal of speech in a probabilistic context. The variants of the n-gram model which uses a probabilistic approach for the occurrence of words are used in the large-vocabulary speech recognition systems [6]. Researchers have investigated various techniques to improve the robustness of speech recognition systems against variability in between training and testing environments, triggered by background noise, the individuality of voice, microphones, transmission channels, reverberation in the room, and so on. In Japan, a 5-year national project entitled "Spontaneous Speech: Corpus and Processing Technology" was carried out. As a result, the largest spontaneous speech corpus in the world, "Corpus of Spontaneous Japanese (CSJ)", consisting of about seven million words, which is 700 hours of speech, was developed, and various new techniques were investigated. These new techniques include robust acoustic modeling, sentence boundary detection, pronunciation modeling, an adaptation of acoustic and language models, and automated speech review [7].

Machine learning and artificial intelligence have assisted in the development of speech recognition. In the modern area, for high-resource language, the deep learning-based approach has become the state-of-the-art approach. In 2012, the ASR dominance of the Deep Neural Network (DNN) started, showing that feed-forward DNN outperforms GMM in the task of estimating context-dependent HMM emitting probabilities [8]. Graves and Jaitly [9] created a voice recognition system that transcribes audio data straight into text without the use of a phonetic intermediate representation, eliminating the necessity for HMM. The method is based on a combination of the recurrent neural network architecture of the deep Bidirectional Long Short-Term Memory (Bi-LSTM) and the objective function of the Connectionist Temporal Classification (CTC). The direct translation of an audio signal into a grapheme series enables the framework to be easily extended to new languages. After the launch of DNN, WER has reduced approximately 25 percent regarding the best previous systems [10]. Attention-based Transformer networks began to be used in speech recognition with the development of attention-based networks in various sectors of AI and NLP [11, 12].

Nepali Speech recognition has been used for the past couple of decades. Nepali Speech recognition has been following the development path followed by English ASR. Preliminary Nepali ASR tools were HMM-based Nepali Speech recognition tools that had been

used to identify 10 distinct Nepali words. Later ASR is built used HMM, Java speech grammar format in the acoustic model of Sphinx-4 [13]. Some research regarding LSTM-CTC being used for Nepali Speech recognition has already been conducted. Nepali Speech recognition CNN for spatial information extraction, GRU to construct an acoustic model, and CTC for decoding have also been developed and promise to provide a very good model that could deliver good WER [14].

Another very good Speech-To-Text (STT) system based on CNN and its variant had been discussed in recent papers [15, 16]. The effectiveness of several models had been assessed and compared in this paper. In order to decode Nepali letters effectively and with the least amount of mistakes, a beam search decoder has been employed in this research. ASR performed on Nepali Speech corpus from the OpenSLR portal resulted in an overall CER of 23.72%. These were merely the beginning steps of speech or sentence recognition. They were limited to a fixed number of words and statistically determined the high-probability words.

3. **Methodology.** An automatic speech recognition system requires a series of steps and processes. Major steps in the ASR are data preparation, data pre-processing, feature extraction, model building, training, testing, fine-tuning hyper-parameters and validation. The clean dataset is a preliminary requirement to use any machine learning algorithm. The clean dataset in automatic speech recognition means the speech corpus is clear without excessive noise and the text transcript transcribes the speech exactly. Dataset preparation is one of the major tasks during the research work. Dataset preparation involves selecting speech audio that is clear and speech can be recognized that has less noise. Not only the quality of audio but audio that represent the exact speech corresponding to the transcript must be selected. After the dataset preparation, data cleansing is performed on the dataset. This involves removing special characters like exclamation marks, question marks, and single & double quotes in the transcript corresponding to the speech audio.

Figure 1 shows the steps that have been proposed in this research work. Feature extraction is another step performed after audio files are pre-processed. This involves extracting audio features hidden in the speech audio. Feature extraction can be done using various techniques. Extracting the sound level at a different time is one of the primitive audio feature extraction techniques. Recently, the most commonly used audio feature extraction techniques include spectrogram extraction, MFCC extraction, etc. The audio feature has been extracted using STFT of the audio and then visual features are extracted using CNN technology.

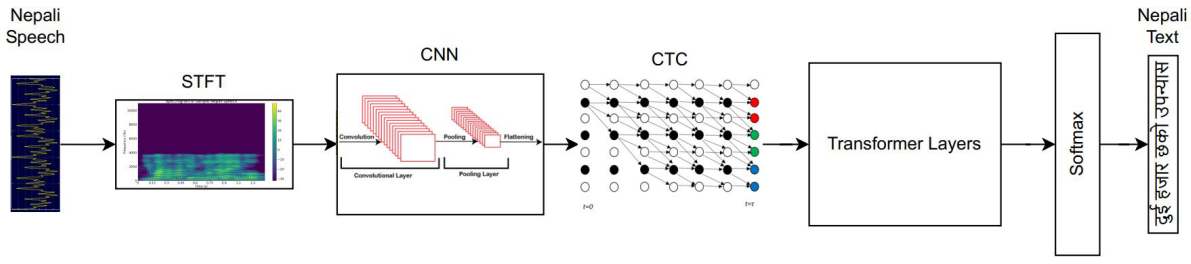

FIGURE 1. System block diagram of Nepali Speech recognition system

Features are extracted from the audio corpus and these features are passed to the ASR model to detect the transcripts corresponding to the speech. The visual feature of speech segment represented by STFT is extracted using CNN. The visual features extract and true text value is aligned using CTC which solves the text and speech alignment issue that

may arise due to variation in speech rate of different speakers. The self-attention layer and feed-forward neural networks are combined together to form encoders and decoders of the Transformer network. Encoder has been used to generate high-level representation among the input audio sequences. This generates a high-level representation of audio features. Decoder has been used to generate target character sequences. The decoder has modeled the data as a conditional language model.

3.1. **Nepali language and Nepali grammar structure.** The Nepali language is the national language of the country Nepal. It is used all over Nepal in the educational system, and public administration, as a language for mass communication, and for general communication language in major parts. Devanagari Script is used to represent the Nepali language in written form. The Devanagari Script is also used to represent other languages like Hindi, Marathi, and Sanskrit. In the Nepali language, there are 12 vowels and 33 consonants. The nature of Nepali script is phonetic, and the pronunciation therefore closely resembles that of writing system. Nepali script is composed from left to right. There is no provision for capital and small letters in script. Alphabets are written in two separate language groups, namely the consonants, and vowels [17]. Nepali Speech recognition is challenging because of variations that can be brought in Nepali text writing. Combination consonants produce varied sounds when combined with 12 different vowels. This creates difficulty in detecting text from a large number of possible combinations of Nepali text components. Therefore, the challenge of Nepali Speech recognition is first training model with texts other than computer known English alphabet and later finding the combinations of vowels and consonants correctly from speech data.

3.2. **Dataset collection.** Automatic speech recognizers are one of examples of supervised learning that required labeled dataset. Selecting the proper dataset for training ASR is a major task while building a good ASR model. Two different datasets have been taken from two different sources and with the different contexts of contents of speeches.

3.2.1. *Nepali Sports News corpus.* Nepali Sports News corpus which is a speech corpus spoken by 3 different speakers has been used to train the Transformer network. The speech content is related to Nepali Sports News. It comprises a total of 2768 speech utterances, which together make up a Nepali Speech corpus of 1 hour 43 minutes. This dataset has been collected from students of the Institute of Engineering Bachelors in an attempt to build a better Nepali Speech corpus. The contents have been scrapped from the popular online Nepali News portals. This is not a standard dataset but this research work has used this dataset to evaluate the model performance on Nepali Sports News context-specific dataset.

3.2.2. *OpenSLR Nepali Speech corpus.* For general purpose speech recognition speech corpus, the Nepali Speech corpus from OpenSLR named "Multi-speaker TTS data for Nepali (ne-NP)" (SLR43) has been used to train the Transformer network. The dataset consists of speech utterances of texts related to varied subject matters. The subject matter of the Nepali Speech in the OpenSLR speech corpus ranges from Nepali art, literature, and entertainment to international affairs and different subjects occurring in other countries. This is the standard dataset available in the OpenSLR speech corpus portal. It contains about 2064 utterance spoken by 18 different female speakers. All speech utterances sum up to speech of a duration of 2 hours and 47 minutes.

3.3. **Data pre-processing.**

3.3.1. *Audio pre-processing.* Before feeding the audio data to models for pre-training, data should be pre-processed with the following steps as per need.

1) Segment the audio into 30 seconds (maximum) each.
2) The background music should be then removed from the audio.
3) The silence contained in the audio should be then removed by specifying a threshold decibel value.
4) The audio needs to be sampled at 16 KHz and any stereo stream should be brought down to a mono channel.

3.3.2. *Text pre-processing.* The data from OpenSLR used for fine-tuning contains text with punctuation marks and numbers. Since the CTC algorithm maps time stamps that represent units of sound to characters, we cannot use digits as characters in our vocabulary. Instead, we must first convert those numbers to words in our transcript. For this, we followed the following steps for the conversion and cleaning.

1) We first mapped all Nepali digits in our corpus to English and then made a list of all numbers in our corpus using regex.
2) After that, we used a script to get Nepali words corresponding to all those numbers and saved the mapping from Nepali numbers to words for every number in the corpus in a key-value pair data structure.
3) Then, we used regex to replace all the numbers with corresponding word mapping.
4) We also dropped all the punctuation marks from our text.

3.4. **Feature extraction.** For speech feature extraction, speech files have been segmented into a fixed-sized window of about 20 ms which is converted into a spectrogram as a frequency domain representation. STFT has been used to convert the time domain Nepali Speech audio corpus to the frequency domain over the windowed period. The converted frequency domain speech is now represented as a frequency spectrum. Visual information from the STFT signal is extracted using the visual information extraction technique of convolutional neural networks.

Therefore, the audio features are generated from the spectrogram and passed to a higher layer of positional encoding to encode the positional value of the audio features.

3.5. **Model architecture.** The model for Nepali Speech recognition consists of encoders and decoders compromising of Transformers. Auxiliary components like character encoding, positional encoding, projection, and down-sampling help the encoder and decoder portions of the Transformers.

The encoder is used to encode the input audio speech and represent it in terms of code. While the decoder is used to engrave the text representation over the encoding generated by the encoder.

Encoder and decoder are two major components of ASR using the Transformer model. Encoder has been used to generate high-level representation among the input audio sequences. This generates a high-level representation of audio features. Decoder has been used to generate target character sequences. The decoder has modeled the data as a conditional language model. Previously generated tokens by the decoder along with a high-level representation of speech sequence generated by the encoder are used to decompose probabilities of the sequence of discrete tokens to the ordered product of distributions. The basic components used in encoders and decoders are not other than neural networks.

These neural networks map the relationship between input and output sequences for various time steps. For automatic Nepali Speech Recognizer, speech audio signals are

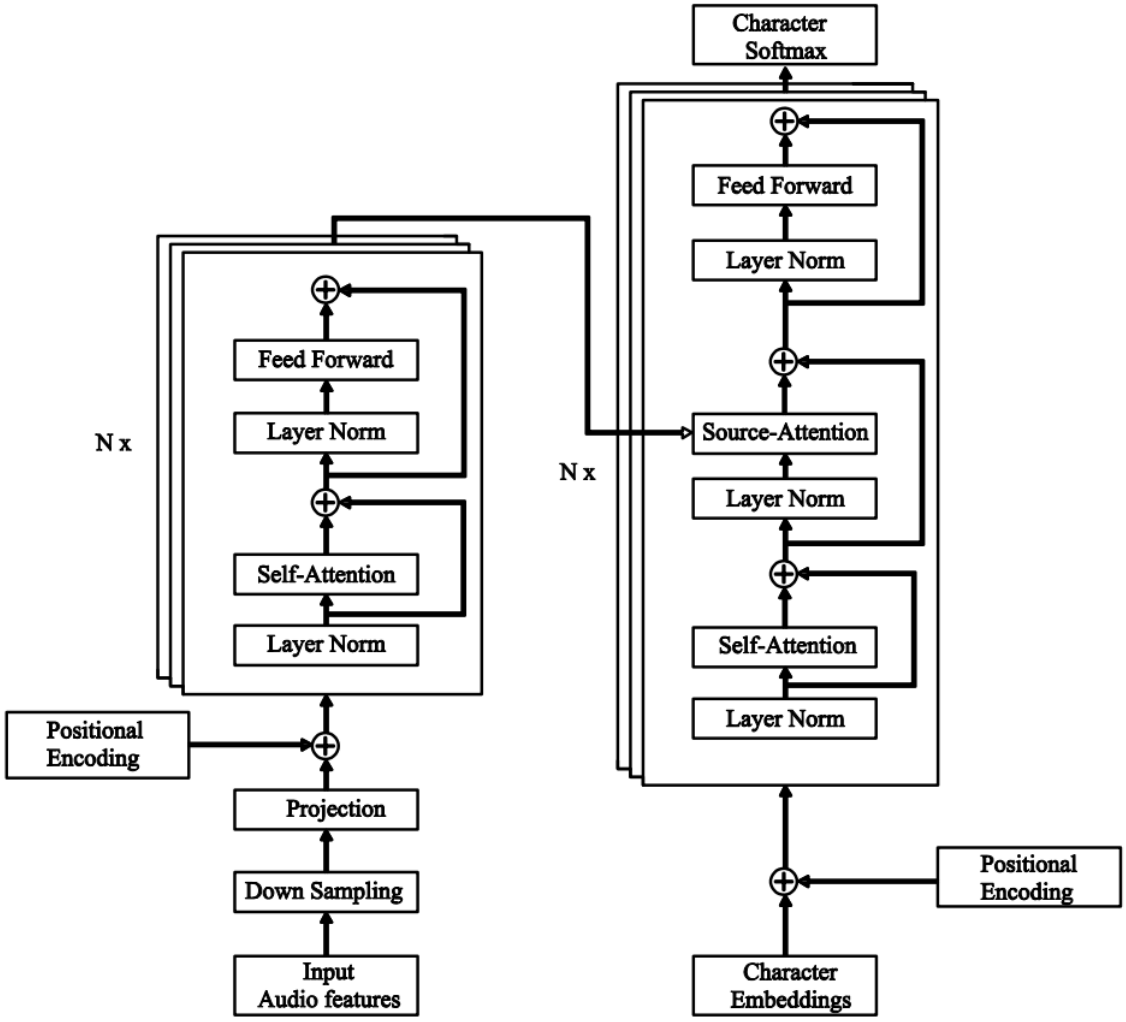

FIGURE 2. A Transformer for Nepali acoustic features to Nepali character-level transcription

the input sequence while target text transcripts are the out sequence. The decoders are required to focus on particular components of the sequence obtained as high-level encoder representation. During the speech, recognition attention is provided to specific time series audio features which represent the transcribed text being considered.

Attention or multi-head attention is used as a fundamental component in the model based on Transformers which has eradicated the use of legacy recurrent networks.

The input layer comprises the layer that is responsible to take in pre-processed feature parameter and pass it to the higher layer for further processing. Earlier models used 8 to 13 cepstral coefficient for feature extraction of the human voice. Similarly, 13 cepstral coefficients are taken as the feature input to DNN for Speech recognition.

Instead, MFCC features extracted from CNN of STFT have been used as input feature for the model. STFT converts the time-domain audio speech to the frequency domain for time-windowed audio speech. Feature from time-windowed frequency spectrum has been extracted using convolutional neural networks.

3.6. **Connectionist temporal classification.** Time sequence data are trained as frame-wise classifiers in which the training dataset has a target label for every frame. Similarly, speech recognition has a similar situation where we need to match unequal frame sizes and

character labels. The same number of character labels might result in varying numbers of speech frames depending upon the reading speed of the reader. This makes the alignment of speech frames and character labels very difficult. CTC has helped in similar scenario requiring synchronization between text frames and speech in subtitle recognition [18]. CTC loss is an error function that helps to train speech to text recognizers like RNN or LSTM over time sequence data that do not have alignment among input speech frames and target character labels.

The typical operation of the CTC algorithm has been depicted in Figure 3. Characters hidden in the different audio frames are extracted after mapping the corresponding characters and the gap between character frames. Speech recognition, scene text recognition, sign language recognition, and video segmentation are some of the activities that CTC can help with. CTC is familiar with the current likelihood calculation's summation of all feasible path probabilities. The main idea of CTC is to employ intermediate label representation so that no output label can be detected by label repetitions or blank label occurrences. The forward-looking algorithm not only effectively quantifies the CTC loss, but also forecasts targets for each frame and assumes that the targets are conditionally independent.

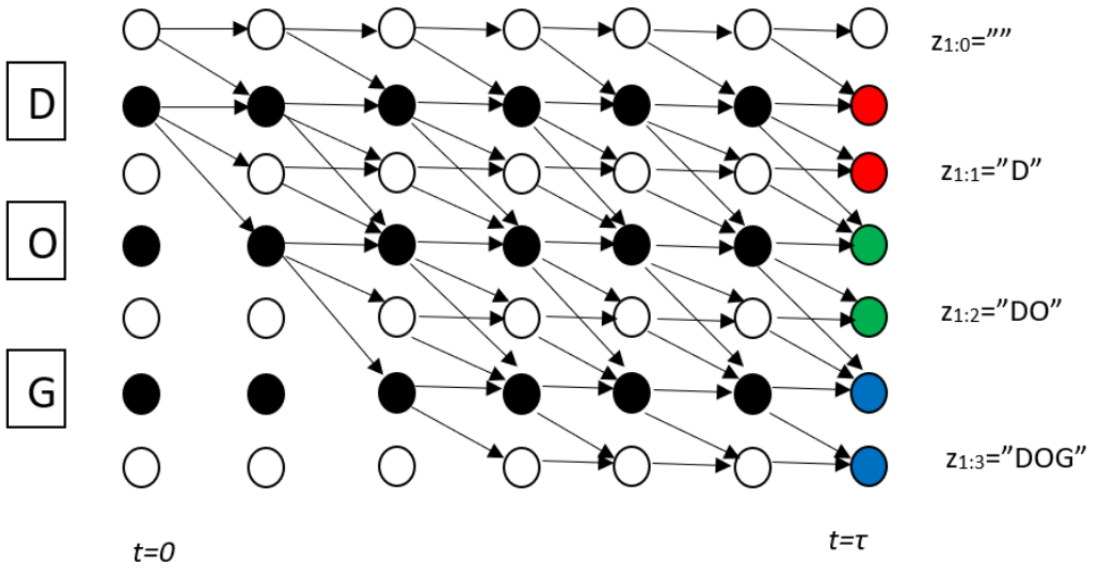

FIGURE 3. CTC algorithm

3.7. **Training with CTC.** The loss value is required to calculate given pairings of speech and ground-based truth texts in order to train the model. The loss is calculated by adding all of the scores from all of the potential ground truth text alignments, regardless of where the text appears in the statement. The score for one alignment is calculated by multiplying the character scores together. There are a total of 129 Unicode characteristics that can represent Nepali Devanagari characteristics but 67 characters in the Nepali language can represent sound in the form of text. Decoding using the CTC algorithm can be done efficiently and promptly in two steps:

- Best path is calculated by taking the most likely character;
- The duplicate characters are eliminated and then all the blanks are removed from the path.

The character set is indexed from 0 to 66 representing '*ka*' as 0 and increases accordingly. Now the output of the dense layer is fed to the SoftMax layer.

'क', 'ख', 'ग', 'घ', 'ङ', 'च', 'छ', 'ज', 'झ', 'ञ', 'ट', 'ठ', 'ड', 'ढ', 'ण',

'त', 'थ', 'द', 'ध', 'न', 'प', 'फ', 'ब', 'भ', 'म', 'य', 'र', 'ल', 'व', 'श', 'ष',

'स', 'ह', 'अ', 'आ', 'इ', 'ई', 'उ', 'ऊ', 'ए', 'ऐ', 'ओ', 'औ', ' ा', ' ि',

' ी', ' ु', ' ू', ' ृ', ' े', ' ै', ' ो', ' ौ', ' ्', ' ँ',

'०', '१', '२', '३', '४', '५', '६', '७', '८', '९','।', ' '

FIGURE 4. 67 Devanagari characters

3.8. **Model evaluation.** The speech recognition model's main function is to accurately detect the texts corresponding to the speech audio input received. Speech recognition models can be evaluated using the error rates displayed at various levels. The evaluation techniques used in the speech recognition model involve three different types of errors. These errors are

- **Substitution error (S):** It measures the character or words that are misspelled with reference to ground truth;
- **Deletion error (D):** It measures the character or words that are lost or missing with reference to the ground truth;
- **Insertion error (I):** It measures the character or words that are incorrectly included over the ground truth.

3.8.1. *WER (Word Error Rate).* The Word Error Rate (WER) is a standard indicator for assessing a voice recognition system's performance. WER is based on the Levenshtein distance, where it works at the word level rather than the phoneme level. If W represents no. of words correctly predicted and N represents no. of words in ground truth ($N = S + D + C + W$), then the word error rate can be represented mathematically as

$$WER = \frac{S + D + I}{N} \tag{1}$$

3.8.2. *CER (Character Error Rate).* Character Error Rate (CER) is also calculated using the Levenshtein distance concept, which counts the minimal number of character-level operations required to transform the ground truth text into speech recognition output.

If C represents no. of characters correctly predicted and N represents no. of characters in the ground truth, then the CER can be represented mathematically as

$$CER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C} \tag{2}$$

The percentage of characters in the reference text or ground truth that inaccurately predicted speech recognizer output is the result of Equation (2).

The ASR model's performance improves as the CER value decreases. The best automatic speech recognizer is characterized by the lowest CER. The highest and ideal CER obtainable is 0%.

4. **Results and Analysis.** A speech recognition model with Transformer network using a self-attention mechanism has been built. The model has been successfully trained over Nepali Speech corpus and the model has shown the good performance of Nepali Speech recognition.

The Transformer networks are very complex models that require huge computing and memory resources. Earlier models using RNN, LSTM, and GRU could be implemented in the computing resources available in simple laptop configuration but attention-based Transformer networks require huge computing resources.

A speech recognition model has been built which is able to convert the speech data to the transcript of the speech data. This speech recognition model consists of Transformers as the major component which helps to implement the attention mechanism.

The Transformer networks consist of encoders that encode the input features in special representation. Similarly, the decoder helps to map the output transcript with the encoded representation of input features.

Transformer Components: A typical model of a self-attention-based Transformer network was built using attention-based Transformers. The components of the Transformer are as

- Positional Encoder
- 4 Encoder Layers
  - Multi-Head Attention layer
  - SoftMax layer
  - Layer Normalization
  - Positional Feed forward Network
- 4 Decoder Layers
  - Multi-Head Attention layer
  - SoftMax layer
  - Layer Normalization
  - Positional Feed forward Network

Model Specifications: The model was trained with the following specifications:

- Batch-size: 12
- Initial learning rate: 0.00001
- Number of layers: 4
- Number of heads: 8
- Model dimension: 512
- Dropout rate: 0.1

The experimental results obtained during different experiments have been illustrated in the following sections.

4.1. **Experiment 1.** The ASR model built using the attention-based Transformers has been trained by OpenSLR SLR43, i.e., ne_np_female High-quality TTS data for Nepali spoken by 18 different female speakers. Specifications of this Nepali corpus are as

- Total number of utterances: 2064
- Total duration of speech: 2 hours and 47 minutes
- Training duration: 70 hours

The OpenSLR dataset was divided into a training set, validation set, and test set in a ratio of 60 : 20 : 20. Training set has been used to train the model, the validation set has been used to tune hyper-parameters and the test set has been used at last to test the performance of Nepali Speech recognition model. While training the automatic speech recognition using self-attention based Transformer by OpenSLR Nepali dataset training and validation loss profile are seen as in Figure 5. The figure shows the value of training and validation loss observed at different training epochs indicated by two different colors.

The loss profile shows training and validation losses observed at different epochs while training the Transformer model. Both training and validation loss reduction is abrupt

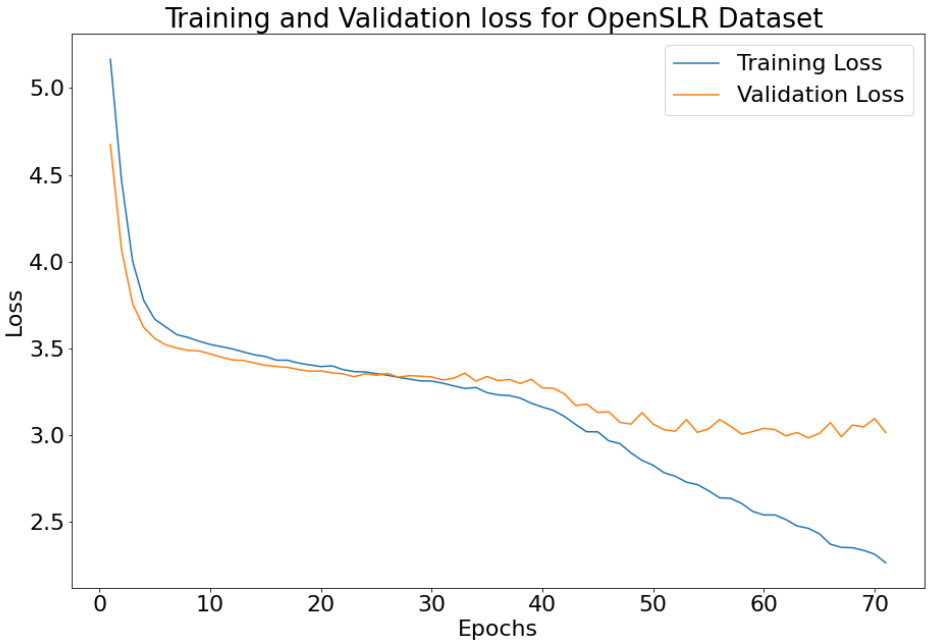

FIGURE 5. Training loss and validation loss for OpenSLR dataset

from the beginning to the epoch of 10. From epoch 10 losses gradually decrease till epoch 36. However, just after a few more epochs, a special situation is observed where training loss continues to decrease while validation loss stops decreasing and starts increasing. This behavior depicts the phenomenon of over-fitting. An over-fitting phenomenon observed after a particular epoch, we take a model between 36 and 40 as the best model. Finally, the test was performed on the best model built on the test dataset segregated as the test portion from the whole dataset. The results observed showed a CER of 55.65% and a WER of 95.32%.

4.2. **Experiment 2.** In this experiment, the automatic Nepali Speech recognition using a self-attention based Transformer model has been trained by the Nepali Speech corpus based on sports news. It consisted of 2768 utterances by 3 different speakers. The total speech duration of speech is one hour and 43 minutes. Model loss stabilized at about 36 epochs. Train and validation losses are observed as depicted in Figure 6.

- Total number of utterances: 2768 (by 3 different speakers)
- Total duration of speech: 1 hour 43 minutes
- Training duration: 75 hours

The loss observed in this Nepali Sports News dataset is less as compared to the loss observed in the OpenSLR general context dataset. Figure 6 shows training loss and validation loss start to decrease abruptly from the start of training till the 25th epoch. Then losses gradually decrease with the increase of epoch number. After epoch number 40, training loss seems to be stabilized at about 0.2 while validation loss stabilizes at about 0.5. The co-linear position of training loss and validation loss suggest the proper fitting of Nepali Sports News dataset to the model. However, training loss is a bit higher than validation loss.

Comparison of performance of model trained on OpenSLR dataset and the same model trained on specific context based Nepali Sports News dataset with CNN-RNN model trained on OpenSLR dataset is shown in Table 1.

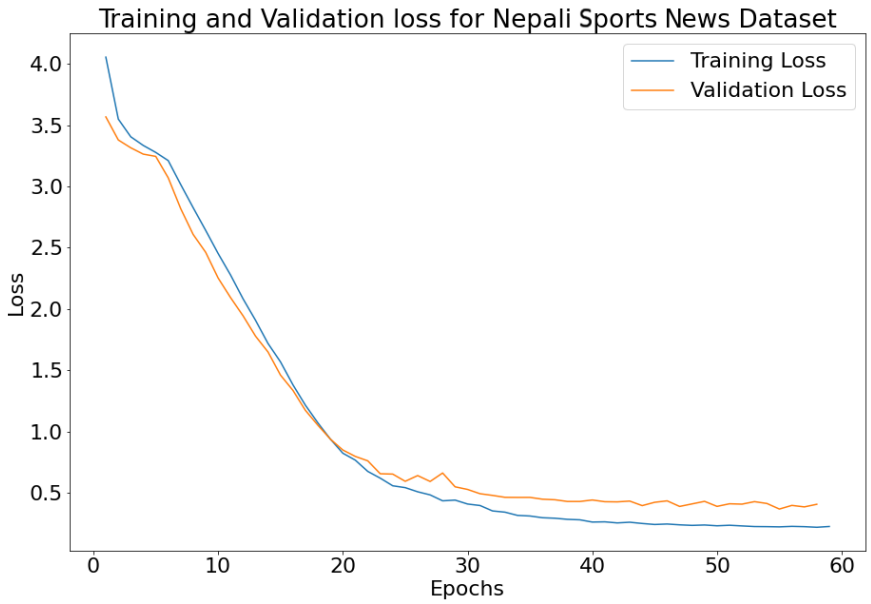

FIGURE 6. Training and validation loss for Nepali Sports News dataset

TABLE 1. CER and WER comparison of datasets based on different contexts

Model and dataset	**CER(%)**	**WER(%)**
CNN-RNN (OpenSLR dataset) [15]	23.72%	–
Acoustic attention model (OpenSLR dataset)	55.65%	95.32%
Acoustic attention model (Nepali Sports News dataset)	19.55%	27.59%

CER of acoustic attention model trained over Nepali Sports News dataset is 19.55% which is one of best results obtained. However, the same model trained on general context Nepali Sports News dataset resulted CER of 55.65% which is worse than the CNN-RNN based model trained over the same OpenSLR dataset which resulted CER of only 23.72%.

This result on the Nepali Sports News dataset is very good compared to the model trained over the OpenSLR general context Nepali Speech corpus. Training an attention-based speech recognition model that detects corresponding texts to audio features by two different types of datasets has given different results that could tell us even more about the attention based speech recognition models. The OpenSLR dataset is a general dataset that contains speech data related to a wide range of topics including art, literature, international and national news, and general knowledge related to Nepal and all over the world. This general context of the dataset has prevented the model from obtaining good results as the model encountered greater loss during the training phase. Performance CNN-RNN models are seen better in general context OpenSLR datasets as compared to basic attention-based speech recognizer.

However, while training the second dataset of the Nepali Sports News dataset which is Nepali Sports related news dataset, as this dataset is based on a particular context of Nepali Sports News attention-based speech recognition model could focus well and give better results in predicting the text for the test dataset.

4.3. **Experiment 3.** The Transformer used in automatic speech recognition using self-attention based Transformer networks consists of numerous layers of encoders and decoders. These layers of encoder and decoders help in attending different features in the input provided and help in generating more accurate automatic speech recognizer results.

Therefore, in an attempt to explore the performance of automatic speech recognizers using self-attention based Transformer networks, an experiment was performed. The experiment consisted of evaluating the performance of attention-based Transformer models with change in the number of layers of the Transformer model.

Table 2 shows the performance of automatic speech recognition using self-attention based Transformer models with different layers of Encoder-Decoders. Increasing layers of Transformer, the complexity of the model has also been increased along with the computational time and memory usage of a computer during training to the Transformer model.

TABLE 2. Performance comparison of Transformer network with a different number of layers

Model	**CER(%)**	**WER(%)**
CNN-RNN [15]	23.72%	–
Acoustic attention model (4Enc-4Dec)	55.65%	95.32%
Acoustic attention model (8Enc-8Dec)	33.03%	62.42%
Acoustic attention model (12Enc-12Dec)	25.37%	58.25%

CER and WER have increased with the increase in a layer of Encoder-Decoder in Speech Recognizer using self-attention based Transformer model. CER of 55.65% and WER of 95.32% have been observed with 4 layers of the Encoder-Decoder in Nepali Speech Recognizer using self-attention based Transformer model. The layers of the Encoder-Decoder have now increased by 4 layers. CER of 33.03% and WER of 62.42% have been observed for Nepali Speech Recognizer using self-attention based Transformer model with 8 layers of Encoder-Decoder.

The performance of the Nepali Speech Recognizer using self-attention based Transformer network is still not satisfactory so in an attempt to attain higher accuracy layer is further increased by 4 units. CER of 25.37% and WER of 58.25% have been observed with 12 layers of Encoder-Decoder in Nepali Speech Recognizer using self-attention based Transformer model. The added layers of the Transformer model have added complexity to the model and the current experimental setup cannot handle the training of the model. The 12-layer Nepali Speech Recognizer using self-attention based Transformer model still cannot outperform CNN-RNN Nepali Speech Recognizer which resulted in a CER of 23.37% [15].

The number of layers of encoders and decoders certainly increases the performance accuracy of Nepali Speech Recognizer using self-attention based Transformer models. However, the number of layers has been limited by the computational capacity of the computer used. The model could be built with a larger number of layers of encoders and decoders for higher accuracy but we also need to consider the over-fitting phenomenon that may occur due to training a small dataset over very complex models.

4.4. **Experiment 4.** Nepali Speech recognition using self-attention based Transformer models takes audio as input and generates the corresponding transcript for provided audio speech. The different numbers of audio can be fed to the Transformer models that can process different numbers of audio inputs at a time. A batch size of 4 has been used in earlier experiments to train the model. This batch size has been varied and the performance of the model for different batch sizes has been evaluated.

Experiments have been carried out to check the performance of Nepali Speech attention-based Transformer models when batch size is varied. The Transformer model processes the

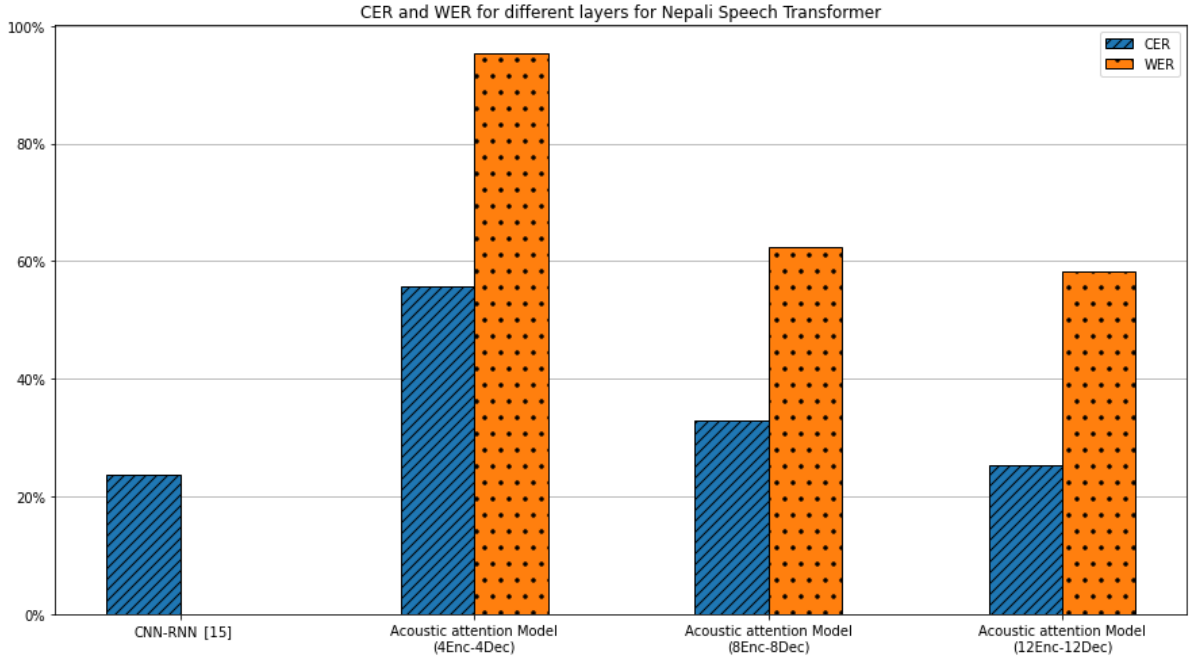

FIGURE 7. Bar chart comparing performance of Nepali Speech Recognizer using self-attention based Transformer with different numbers of Encoder-Decoder layers

input of mentioned batch size at a time. This increases the input size to the Transformer model but reduces the number of iterations. Increasing the batch size increases the number of inputs for a single iteration, causing more memory usage to store and process large input audio features at a single iteration. Limitation of the experiment to increase the batch size because of limited memory resources has limited the experiment up to the batch size of 12.

Table 3 shows the performance of automatic speech recognition using self-attention based Transformer models with different batch sizes input to the Transformer. Increasing input batch size to Transformer, input data size to be processed by the model increases along with the computational time and memory usage of the computer during training Transformer model.

TABLE 3. CER and WER comparison of datasets based on batch size

Model and dataset	**CER(%)**	**WER(%)**
CNN-RNN (OpenSLR dataset) [15]	23.72%	–
Acoustic attention model (Batch size = 4)	25.37%	58.25%
Acoustic attention model (Batch size = 8)	22.42%	48.47%
Acoustic attention model (Batch size = 12)	19.47%	38.69%

CER and WER have increased with an increase in input batch size to Speech Recognizer using self-attention based Transformer model. CER of 25.37% and WER of 58.25% have been observed with 12 layers of the Encoder-Decoder and input batch size of 4 in Nepali Speech Recognizer using a self-attention based Transformer model. CER of 22.42% and WER of 48.47% have been observed for Nepali Speech Recognizer using self-attention based Transformer model with input batch size changed to 8.

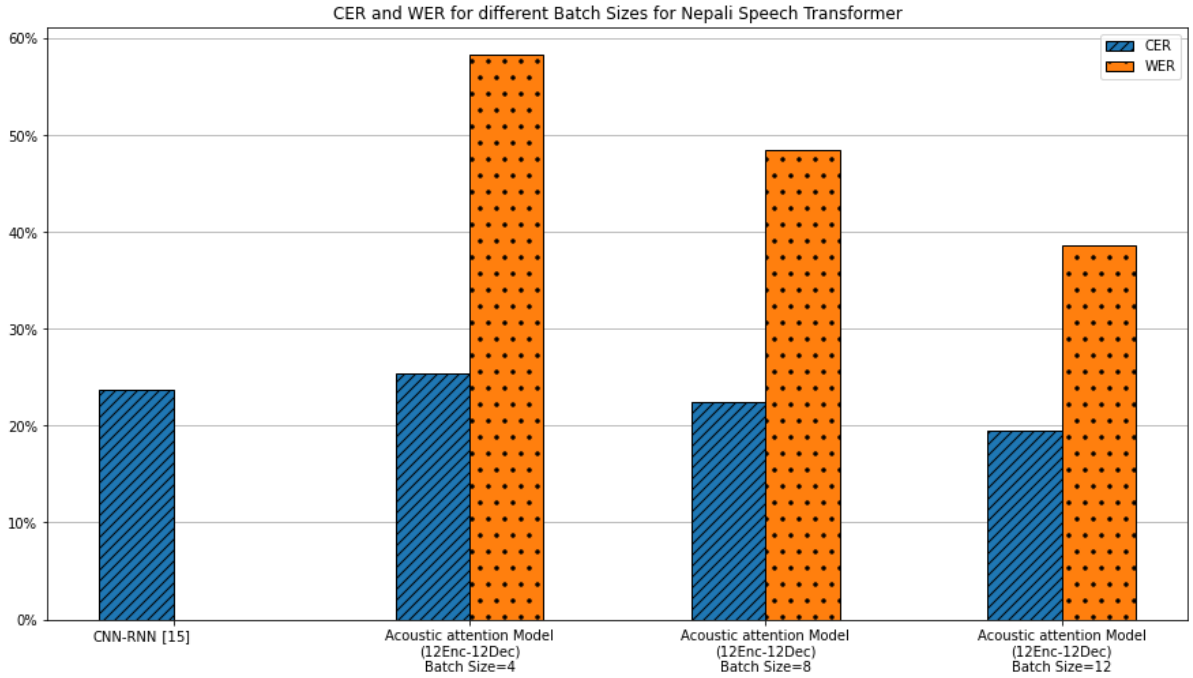

FIGURE 8. Bar chart comparing performance of Nepali Speech Recognizer using self-attention based Transformer with different Batch size

The input batch size has now been increased to 12 to get optimum results from the model. CER of 19.47% and WER of 38.69% have been observed with 12 layers of Encoder-Decoder with input batch size of 12 in Nepali Speech Recognizer using self-attention based Transformer model.

5. **Conclusions.** In this research, Nepali Speech recognition using a self-attention based Transformer network has been proposed. The attention mechanism showed better performance of the Nepali Speech recognition in the context-specific dataset than in the general context dataset. CER of 55.65% and WER of 95.32% were observed for the OpenSLR general context dataset whereas CER of 19.55% and WER of 27.59% were observed for the context-specific Nepali Sports News dataset. CER of 19.47% and WER of 38.69% have been observed while testing over general context Nepali Speech corpus from OpenSLR after optimizing Transformer layers and input batch size. The performance of the Nepali Speech Recognizer using attention-based Transformers was improved by increasing encoder-decoder layers and batch size.

In future work, we are interested in adding a word dictionary in the Nepali language and determining correct words from the pre-defined word list. This would certainly help to improve the speech recognition model and improve the usability of speech recognizer in the real world as a reliable machine learning tool.

Acknowledgment. This work has been supported by the University Grants Commission, Nepal under a Faculty Research Grant (UGC Award No. FRG-76/77-Engg-1) for the research project "Preparation of Nepali Speech Corpus: Step towards Efficient Nepali Speech Processing".

REFERENCES

[1] H. Dyoniputri and Afiahayati, A hybrid convolutional neural network and support vector machine for dysarthria speech classification, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.111-123, 2021.

[2] C. Prajapati, J. Nyoupane, J. D. Shrestha and S. Jha, Nepali speech recognition, *Kathmandu. DOECE*, 2008.

[3] K. H. Davis, R. Biddulph and S. Balashek, Automatic recognition of spoken digits, *The Journal of the Acoustical Society of America*, vol.24, no.6, pp.637-642, 1952.

[4] S. Furui, Speaker-independent isolated word recognition using dynamic features of speech spectrum, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol.34, no.1, pp.52-59, 1986.

[5] L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, *Proceedings of the IEEE*, vol.77, no.2, pp.257-286, 1989.

[6] F. Jelinek, The development of an experimental discrete dictation recognizer, *Proceedings of the IEEE*, vol.73, no.11, pp.1616-1624, 1985.

[7] S. Furui, 50 years of progress in speech and speaker recognition research, *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol.1, no.2, pp.64-74, 2005.

[8] G. Hinton, L. Deng, D. Yu et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, *IEEE Signal Processing Magazine*, vol.29, no.6, pp.82-97, 2012.

[9] A. Graves and N. Jaitly, Towards end-to-end speech recognition with recurrent neural networks, *International Conference on Machine Learning*, pp.1764-1772, 2014.

[10] B. Popović, E. Pakoci and D. Pekar, End-to-end large vocabulary speech recognition for the Serbian language, *International Conference on Speech and Computer*, pp.343-352, 2017.

[11] N.-Q. Pham, T.-S. Nguyen, J. Niehues, M. Müller, S. Stüker and A. Waibel, Very deep self-attention networks for end-to-end speech recognition, *arXiv Preprint*, arXiv: 1904.13377, 2019.

[12] G. I. Winata, S. Cahyawijaya, Z. Lin, Z. Liu and P. Fung, Lightweight and efficient end-to-end speech recognition using low-rank Transformer, *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2020)*, pp.6144-6148, 2020.

[13] A. Kalakheti, K. P. Bhattarari, S. Kuwar and S. Adhikari, *Automatic Speech Recognition for Nepali Language*, Thesis, Tribhuvan University, Nepal, 2013.

[14] B. Bhatta, B. Joshi and R. K. Maharjhan, Nepali speech recognition using CNN, GRU and CTC, *Proc. of the 32nd Conference on Computational Linguistics and Speech Processing (ROCLING2020)*, pp.238-246, 2020.

[15] J. Banjara, K. R. Mishra, J. Rathi, K. Karki and S. Shakya, Nepali speech recognition using CNN and sequence models, *2020 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)*, pp.1-5, 2020.

[16] M. Dhakal, A. Chhetri, A. K. Gupta, P. Lamichhane, S. Pandey and S. Shakya, Automatic speech recognition for the Nepali language using CNN, bidirectional LSTM and ResNet, *2022 International Conference on Inventive Computation Technologies (ICICT)*, pp.515-521, 2022.

[17] B. K. Bal, Structure of Nepali grammar, *Nepali*, pp.332-396, 2004.

[18] G. Sarayut, S. Olarik and P. Pornntiwa, Fusion convolutional recurrent neural networks for Thai and English video subtitle recognition, *ICIC Express Letters*, vol.16, no.12, pp.1331-1339, 2022.

Author Biography

Basanta Joshi received a Doctor of Engineering from Osaka Sangyo University, Japan in 2013. He did both BE (Electronics & Communication Engineering) and ME (Information and Communication) from Institute of Engineering (IOE), Tribhuvan University (TU), Nepal.

Currently, he is working as Assistant Professor at Department of Electronics and Computer Engineering, Pulchowk Campus, Deputy Director at Center for Applied Research and Development and Member at Laboratory for ICT Research and Development (LICT) of IOE. Formerly, he worked as a Coordinator of a Master's in Information and Communication Engineering, he used to work as a Senior Software Engineer in D2Hawkeye and as a Research Consultant at LogPoint. He has been involved as IT/Research Consultant in various national and international projects. He is actively involved in publishing papers in the field of machine learning and its application in big data. He is a member of NEC, NEA, IEEE, ISCA Speech & AEHIN.

Rupesh Shrestha obtained his Bachelor degree of Electronics and Communications Engineering from Purbanchal University and M.Sc. degree in Information and Communication Engineering from Institute of Engineering, Tribhuvan University, Nepal in 2021.

Mr. Rupesh is currently a full-time Telecommunication Engineer at the Nepal Telecom, Nepal. His main research interests include machine learning and data analytics.