Study of Speech Recognition System Based on Transformer and Connectionist Temporal Classification Models for Low Resource Language

Shweta Bansal[1(✉)], Shambhu Sharan[2], and Shyam S. Agrawal[3]

[1] K R Mangalam University, Gurugram, India
s.bansal6281@gmail.com
[2] Indira Gandhi Delhi Technical University for Women, Delhi, India
[3] KIIT Group of Colleges, Gurugram, India

Abstract. Sequence-to-sequence methods have been extensively used in end-to-end (E2E) speech processing for recognition, translation, and synthesis work. In speech recognition, the Transformer model, which supports parallel computation and has intrinsic attention, is frequently used nowadays. This technology's primary aspects are its quick learning efficiency and absence of sequential operation, unlike Deep Neural Networks (DNN). This study concentrated on Transformer, an emergent sequential model that excels in applications for natural language processing (NLP) and neural machine translation (NMT) applications. To create a framework for the automated recognition of spoken Hindi utterances, an end-to-end and Transformer based model to understand the phenomenon classification was considered. Hindi is one of several agglutinative languages, and there isn't much information available for speech/voice recognition algorithms. According to several research, the Transformer approach enhances the performance of the system for languages with limited resources. As per the analyses done by us, it was found that the Hindi-based speech recognition system performed better when Transformers were used along with the Connectionist Temporal Classification (CTC) models altogether. Further, when a language model was included, the Word Error Rate (WER) on a clean dataset was at its lowest i.e., 3.2% .

Keywords: Connectionist Temporal Classification · CTC model · Low-resource language · Speech recognition · Transformer

1 Introduction

People are integrating cutting-edge information and digital technology more and more into their daily lives [1]. Such technologies include automatic speech recognition (ASR), pictures recognition, and speech synthesis. In particular, voice-based solutions seem to be frequently employed in robotics, telecommunications, as well as other industrial fields [2, 3]. One approach to communicating with technology is through speech recognition.

S. R. M. Prasanna et al. (Eds.): SPECOM 2022, LNAI 13721, pp. 56–63, 2022.
https://doi.org/10.1007/978-3-031-20980-2_6

With speech recognition technology, single words or text passages may be recognized and converted into instructions or word sequences.

Traditional speech recognition technologies rely on lexicons, linguistic models, and acoustics, as depicted in Fig. 1 [4]. These systems' components were trained independently, making it difficult to configure them. This decreased the effectiveness of employing these systems. Deep learning has increased the efficacy of speech-to-text systems. GMM was replaced by artificial neural networks for acoustic modelling, which enhanced the outcomes of several research projects [5–7]. However, one amongst the widely used and popular model for continuous speech recognition is the HMM-DNN architecture.

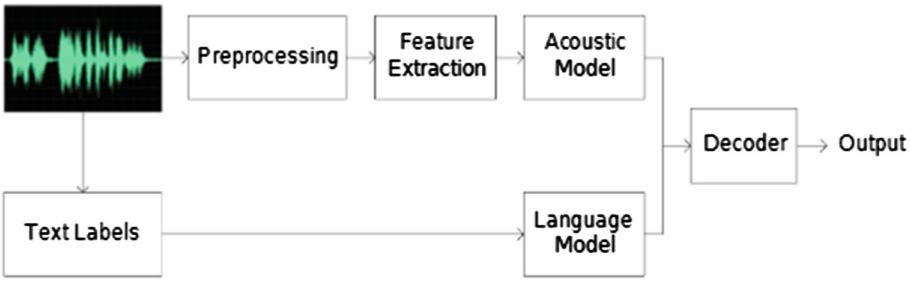

Fig. 1. The traditional ASR architecture [8].

The end-to-end (E2E) system offers knowledge of auditory signals mostly as label sequences with no intermediary stages, necessitating no further output processing, making it simple to implement. The primary issues associated with the description of such approaches, the gathering of a reasonably big database of speech with proper transcription, and the accessibility of highly efficient machinery must be resolved to improve the performance of E2E systems [9]. These problems must be determined to deploy speech recognition systems and other deep learning systems successfully. E2E methods may also significantly boost recognition quality by learning or training from a lot of training data. The Convolutional Neural Network (CNN) as well as improved Recurrent Neural Networks (RNNs) are used in aforementioned E2E models. The RNN models that are used produce a hidden state series with respect to network's prior hidden state by performing computations on position of characters of input as well as output data. With more extended input data sequence and a longer training time, the technique doesn't enable parallelisation of learning in training instances. Another Transformer-based approach that parallelises the learning process and eliminates repeats was suggested in [10]. This model also leverages internal attention to find relationships between the input and output data. Combining an E2E model, such as CTC, along with the Transformer model improved performance of the English and Chinese ASR system to a greater extent, according to earlier studies [11, 12]. It is very much important to mention that the attention mechanism is a typical technique which significantly raises effectiveness of proposed system in speech recognition and NMT domain. Furthermore, the Transformer model speeds up learning by using this attention mechanism. This model has internal alignment that finds an alignment-free representation of the set by aligning every location

in the input sequence. Moreover, large amounts of speech data are needed for training to build such models, that in itself likely to be a challenge for language families with little datasets, such as low-resourced languages including Hindi [13, 14].

The primary objective of our work is to apply models based on Transformer & CTC for Hindi language recognition to increase training data and enhance the accurateness of the ASR for continuous spoken Hindi utterances. The Transformer model was initially developed by the Google Brain team for NMT tasks, later taking the place of RNNs in NLP applications. Recurrence was abolished in this model; instead, signals were created for each statement to indicate the relevance of other sequences for this speech utilising the internal attention process (self-attention mechanism). As a result, the characteristics created for certain assertion are outcome of many sequence-feature linear modifications. The Transformer model is made up of a single massive block that is made up of blocks of encoders and decoders, as shown in Fig. 2. Here, the encoder generates a series of intermediate representations after receiving the feature vectors from the audio signal as input, i.e. $Y = (y_1,\ldots, y_T)$. Furthermore, the decoder duplicates the output sequence $Z = (z_1,\ldots, z_M)$ using the incoming representations. Because the model is autoregressive, each stage outputs the previous symbols before moving on to the next.

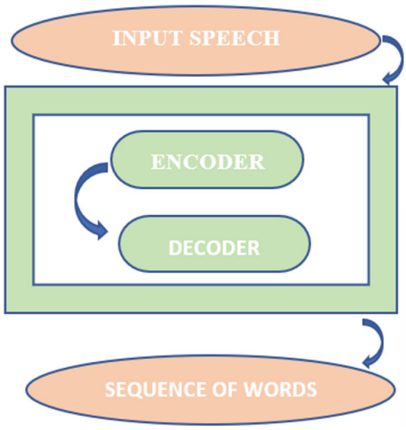

Fig. 2. Basic model of transformer [15].

The overall structure of the paper is presented in different sections, wherein the proposed model is explained in Sect 2, whereas overall experiment and results are presented in Sect 3, where the outcomes are also evaluated. The final section comprises the discussion and conclusions.

2 Proposed Model

Without even pre-aligning the input & output data, recurrent neural networks (RNN) may be trained to identify input voice. The Connectionist Temporal Classification (CTC) is typically utilized for the same [11]. Since straight decoding will not function well, it's

very much required to employ a language model (LM) i.e., an external model comprising probability of different word sequences, in order to get the CTC model to execute well. The method by which words are formed in the Hindi language is also rather varied, which makes it easier to recognise Hindi speech when it is spoken. In this study, the Transformer and CTC models with LM will be used in tandem. When LM CTC is used for decoding, the model converges quickly, cutting down on decoding time and enhancing system performance. When LM CTC is used for decoding, the model converges quickly, drastically reducing on decoding time and enhancing system performance. After receiving the encoder output, the CTC function determines the likelihood of random arrangement among output of the encoder & the output symbol sequence using the formula 1. Here, y is the encoder's output vector, R is a further operator for eliminating repeated symbols and empty spaces, and z is a sequence of anticipated symbols. The formula aids in training the neural network on unlabelled input by utilising dynamic programming to calculate the total of all alignments.

$$P_{CTC}(Z|Encoder(y)) = \sum_{Z \in R} p(z|y)$$

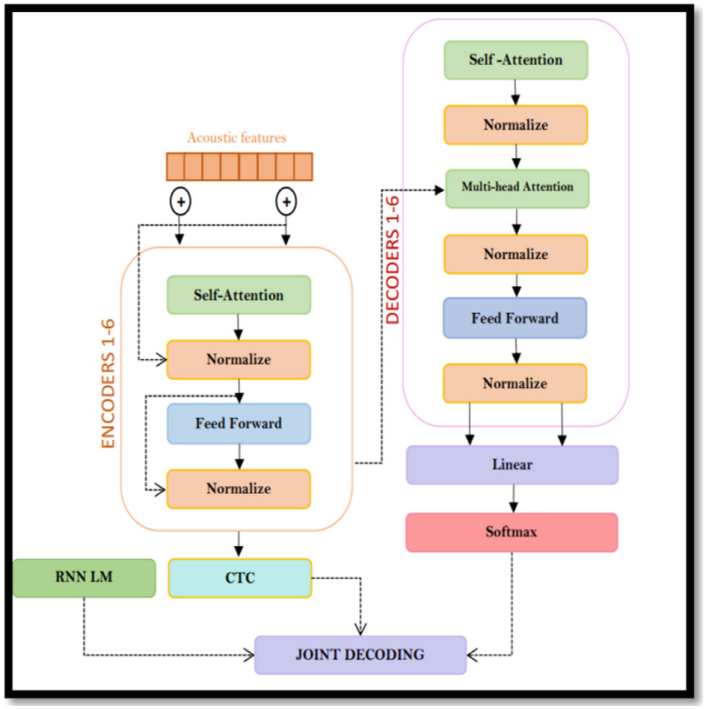

Fig. 3. Overall transformer model used in the experiment [16].

While training with the approach involving multi-task loss, the mathematical expression for combination probabilities based on the negative logarithm was introduced, as

described in [16]. The model that results may be described by the expression 2 where θ is the variable parameter, whose values lies between 0 and 1. Figure 3 depicts the suggested model's structure.

$$P(z|y) = \theta P_{Transformer} + (1 - \theta) P_{CTC}$$

The two criteria were applied to measure the recognition rate of the system for the Hindi language i.e., character error rate (CER), which measures the improperly detected characters, and word error rate (WER) that measures the improperly detected words.

3 Experiment and Results

To train the Transformer model, Transformer model with CTC not having the language model in first case and having the language model in second case, it was ascertained to consider the 300 h of spoken utterance database. This corpus was assembled in the laboratory. The audio files of the spoken utterance database are separated into train and test sections, which are 85% and 15%, correspondingly. The spoken utterance corpus comprises recordings of 200 native Hindi speakers of diverse age groups & genders (Male and Female). The voice recording of the individual speaker took approximately 1 h. Whereas for the text data, sentences containing words of rich phonemes were selected. Text data was collected from various domains like news, defence, general etc. in Hindi language. Students, undergraduates, institute staff members, as well as friends and family members to record the speakers. The voiceovers were recorded over the course of around six months, and to ensure excellent quality, Hindi linguists and linguistics specialists were brought in to examine and review the corpus.

It was important to verify both the accuracy of the transcription of the data as well as the speech data. The construction of a phoneme-level lexicon is not necessary for the speech recognition system; audio and text data are sufficient. After the aforementioned effort, one of the crucial components—a vocabulary foundation for the speech recognition system—was made (11,150 non-repeating words). Repeated words have been eliminated and all recorded messages have been compiled into a single file. After being alphabetized. Wav format was used for the audio data. A single channel has been created out of all audio data. The data was converted into digital form using the PCM technique. 44.1 kHz discrete frequency, 16 bits. The Transformer models were created using the PyTorch toolset.

Table 1. Results obtained for different models.

Models	Character Error Rate (CER)	Word Error Rate (WER)
CTC LM	9.5	18.1
Transformer	8.8	16.7
Transformer + CTC	6.3	12.8
Transformer + CTC LM	3.2	7.9

The Transformer model with CTC produced a CER of 6.3% and a WER of 12.8%, as shown in Table1. The Transformer model with CTC performs well both with and without the usage of the language model. The system became heavier as a result of the incorporation of an external language model, however the CER and WER rates were dramatically decreased by 3.2 and 7.9%, respectively. The directional six-layer Bi-Directional long short term memory is used for the CTC has 256 cells per layer and an interpolation weight of 0.2. At the decode stage, the beam research width is 18. The language model was trained using a created vocabulary base for a speech recognition system and has two 1024-unit LSTM layers. The model has undergone 40 iterations of training.Different models were used in experiments to identify Hindi speech.

As we can see from Table 1, the Transformer and CTC LM model tends to produce optimum results with respect to current database. Furthermore, when compared to other models, as depicted in Fig. 4, the Transformer model with CTC trained and converged considerably faster.

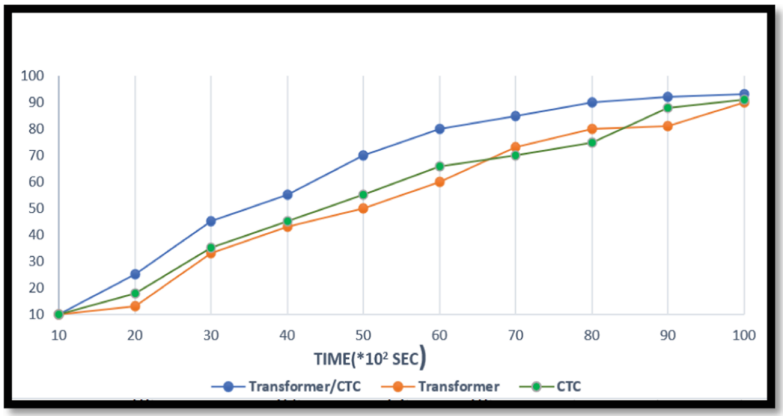

Fig. 4. Comparing curves of Accuracy w.r.t Time during model training.

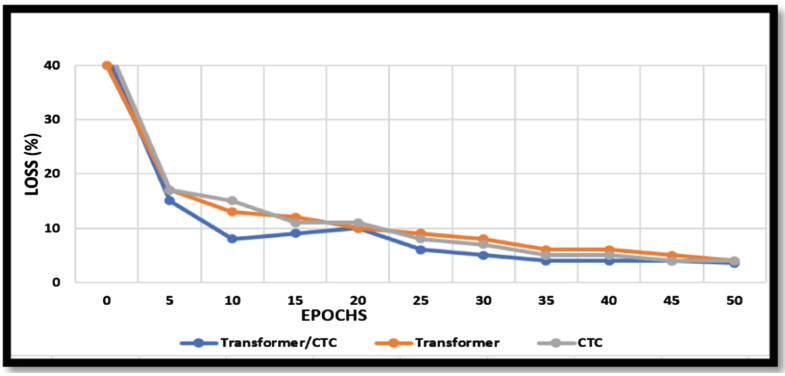

Fig. 5. Loss vs Epoch graph of different models.

Additionally, it was simpler to integrate the final model with LM. The acquired data were aligned with the aid of the CTC. The experiment's findings demonstrate the value of combining the CTC with the E2E language model, and the greatest performance on Hindi dataset was attained. Additionally, our model's performance is often enhanced with the addition of a CTC. In the future, we must increase the size of our speech corpus and enhance CER and WER.

4 Discussion and Conclusion

A DNN-based language model was incorporated into the model to enhance the performance of these measures. In our situation, there is no other option to get decent outcomes than this. Additionally, the quality of recognition can be impacted by adding more trials to a corpus that has a larger volume. However, it's unlikely that just adding more data to the training set would address the issue. The Hindi language has several different accents and dialects. It is impossible to gather adequate information for every situation. Transformer trains the language model better and takes into consideration the complete context, and CTC aids in the model's learning to generate recognition that is best matched with the recording. Further modifications to this architecture are possible for streaming speech recognition.

In this research, the self-attention components of the Transformer architecture for automated identification of Hindi continuous speech were taken into consideration. Although there are many model parameters that need to be adjusted, parallelizing the procedures helps speed up the training process. In terms of character and word recognition accuracy, the combined Transformer + CTC LM model produced an excellent performance in Hindi speech recognition and decreased these numbers by 3.2 and 7.9%, respectively, then utilising them independently. This demonstrates the model's applicability to various low-resource languages.

In order to test the model that has been constructed, it is intended to expand the speech corpus for the Hindi language. Additionally, the Transformer model will need to undergo major changes in order to eliminate word and symbol mistakes in recognition of Hindi continuous speech.

References

1. Anderson, J., Rainie, L.: The positives of digital life (2018), https://www.pewresearch.org/internet/2018/07/03/the-positives-of-digital-life/. Accessed 15 May 2022
2. Deuerlein, C., Langer, M., Seßner, J., Heß, P., Franke, J.: Human-robot-interaction using cloud-based speech recognition systems. Procedia CIRP **97**, 130–135 (2021). https://doi.org/10.1016/j.procir.2020.05.214
3. Rogowski, A., Bieliszczuk, K., Rapcewicz, J.: Integration of industrially-oriented human-robot speech communication and vision-based object recognition. Sensors **20**(24), 7287 (2020). https://doi.org/10.3390/s20247287
4. Sharan, S., Bansal, S., Agrawal, S.S.: Speaker-independent recognition system for continuous hindi speech using probabilistic model. In: Agrawal, S.S., Dev, A., Wason, R., Bansal, P. (eds.) Speech and Language Processing for Human-Machine Communications. AISC, vol. 664, pp. 91–97. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-6626-9_10
5. Seide, F., Li, G., Yu, D.: Conversational speech transcription using Context-Dependent Deep. Neural Netw. (2011). https://doi.org/10.21437/interspeech.2011-169
6. Bourlard, H.A., Morgan, N.: Connectionist Speech Recognition. Springer, Boston, MA (1994). https://doi.org/10.1007/978-1-4615-3210-1
7. Smit, P., Virpioja, S., Kurimo, M.: Advances in subword-based HMM-DNN speech recognition across languages. Comput. Speech Lang. **66**, 101158 (2021). https://doi.org/10.1016/j.csl.2020.101158
8. Yu, C., Kang, M., Chen, Y., Wu, J., Zhao, X.: Acoustic modeling based on deep learning for low-resource speech recognition: an overview. IEEE Access (2020). https://doi.org/10.1109/ACCESS.2020.3020421
9. Perero-Codosero, J.M., Espinoza-Cuadros, F.M., Hernández-Gómez, L.A.: A comparison of hybrid and end-to-end ASR systems for the IberSpeech-RTVE 2020 speech-to-text transcription challenge. Appl. Sci. (2022). https://doi.org/10.3390/app12020903
10. Wang, D., Wang, X., Lv, S.: An overview of end-to-end automatic speech recognition. Symmetry (2019). https://doi.org/10.3390/sym11081018
11. Karita, S., Soplin, N.E.Y., Watanabe, S., Delcroix, M., Ogawa, A., Nakatani, T.: Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration. In: Interspeech-2019, pp. 1408–1412 (2019). https://doi.org/10.21437/Interspeech.2019-1938
12. Miao, H., Cheng, G., Gao, C., Zhang, P., Yan, Y.: Transformer-based online ctc/attention end-to-end speech recognition architecture. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6084–6088 (2020). https://doi.org/10.1109/ICASSP40776.2020.9053165
13. Bansal, S., Agrawal, S.S., Kumar, A.: Acoustic analysis and perception of emotions in Hindi speech using words and sentences. Int. J. Inf. Technol. **11**(4), 807–812 (2018). https://doi.org/10.1007/s41870-017-0081-0
14. Agrawal, S.S., Bansal, S., Sharan, S., Mahajan, M.: Acoustic analysis of oral and nasal Hindi vowels spoken by native and non-native speakers. J. Acoust. Soc. Am. **140**(4), 3338 (2016). https://doi.org/10.1121/1.4970648
15. Bie, A., Venkitesh, B., Monteiro, J., Haidar, M.A., Rezagholizadeh, M.: A Simplified Fully Quantized Transformer for End-to-end Speech Recognition (2019). https://doi.org/10.48550/arXiv.1911.03604
16. Orken, M., Dina, O., Keylan, A., Tolganay, T., Mohamed, O.: A study of transformer-based end-to-end speech recognition system for Kazakh language. Sci. Rep. **12**(1), 8337 (2022). https://doi.org/10.1038/s41598-022-12260-y