Enhancing Performance of End-to-End Gujarati Language ASR using combination of Integrated Feature Extraction and Improved Spell Corrector Algorithm

[1]*Bhavesh* Bhagat and [2]*Mohit* Dua
[1,2]Department of Computer Engineering, National Institute of Technology, Kurukshetra, India

Abstract. A number of intricate deep learning architectures for effective End-to-End (E2E) speech recognition systems have emerged due to recent advancements in algorithms and technical resources. The proposed work develops an ASR system for the publicly accessible dataset on Gujarati language. The approach provided in this research combines features like Mel frequency Cepstral Coefficients (MFCC) and Constant Q Cepstral Coefficients (CQCC) at front-end feature extraction methodologies. Enhanced spell corrector with BERT-based algorithm and Gated Recurrent Units (GRU) based DeepSpeech2 architecture are used to implement the back end portion of the proposed ASR system. The proposed study shown that combining the MFCC features and CQCC features extracted from speech with the GRU-based DeepSpeech2 model and the upgraded or enhanced spell corrector improves the Word Error Rate (WER) by 17.46% when compared to the model without post processing.

Keywords: ASR, MFCC, CQCC, Spell Corrector, Bidirectional Encoder Representations from Transformers (BERT), DeepSpeech2 model

1. Introduction

Automatic Speech Recognition (ASR) has made significant strides in recent years, with the development of more sophisticated models. Hidden Markov Model (HMM) [1] is a statistical model that has been used for speech recognition for several decades and is a prominent approach for ASR. However, HMM-based ASR systems have limitations when it comes to modelling complex linguistic features [2]. Researchers have devised End-to-End (E2E) ASR systems that directly transcribe speech to text without intermediate representations. This research focuses on constructing an ASR system for the Gujarati language using publicly accessible datasets. However, more efficient ASR systems are still required for low-resource languages [3].

The proposed research focuses on constructing an Automatic Speech Recognition (ASR) system for the Gujarati language using publicly accessible datasets. It employs a combination of MFCC and CQCC based integrated front-end feature extraction methodologies [4, 5], as well as Bidirectional Encoder Representations from Transformers (BERT)-based improved spell corrector algorithm and Gated Recurrent Units (GRU)-based DeepSpeech2 model architecture [6, 7]. In comparison to a model without post-processing, the results of the proposed work demonstrate a significant improvement in Word Error Rate (WER). Specifically, the use of integrated features i.e. MFCC and CQCC extraction techniques in conjunction with the GRU-based DeepSpeech2 model and enhanced spell corrector improved WER by 17.46%.

2. Literature Survey

In recent years, E2E ASR systems have emerged as a promising approach, providing a unified framework for all ASR components. E2E ASR systems seek to map an acoustic signal directly to a string of text without requiring intermediate stages such as phoneme or grapheme prediction. Graves et al. (2006) [7] proposed the Connectionist Temporal Classification (CTC) model as one of the earliest E2E ASR models. It has been

demonstrated that CTC-based models are effective for small-vocabulary tasks and low-resource environments, but they may struggle with larger vocabularies and chaotic inputs [8]. Amodei et al. presented an E2E deep learning approach, i.e., end-to-end learning. Deepspeech2 architecture to recognise speech in two significantly distinct languages, such as English and Mandarin Chinese, in 2016 [6]. To develop high performing language recognizers for different languages as dissimilar as English language or Mandarin language, this task required almost no linguistic expertise. We've observed that the procedure is extremely general and easily adaptable to new languages. [9] Raval et al. proposed a method that enhanced the performance or improve the E2E speech recognition system for the language like Gujarati. The hybrid CNN-LSTM model was trained on the Microsoft Speech Corpus dataset using decoding and post-processing methods, and a 5.87% of decrease in WER relative to the base system WER of 70.65% was observed.

In addition, Zang et al. proposed soft-masked misspelt word correction for the Chinese language using the BERT algorithm [10]. As a post-processing technique for the Gujarati language, Raval et al. [9] introduced another BERT-based spelling corrector. This algorithm for correcting misspellings was combined with a pre-trained BERT model to enhance the decoder's output. Dua, M. And Akanksha (2023) proposed an integrated feature extraction technique with a hybrid deep learning model for the ASR of the Gujarati language, which enhances the WER relative to delta-delta features [4]. Motivated by the aforementioned works, particularly [4], [6], and [9], the proposed work improves the performance of E2E Gujarati language ASR by employing an enhanced typo corrector algorithm and GRU-based DeepSpeech2 architecture. The contribution of the proposed work can be summarised as follows:

- The proposed work used integrated feature extraction based method which combines features like MFCC and CQCC to develop the front-end of the proposed E2E ASR system. This function provides a diversity of information that facilitates speech comprehension.
- For the implementation of the downstream acoustic model, we have utilised the GRU-based Deepspeech2 architecture. This model employs Bidirectional GRU units, which are superior to other hybrid CNN-LSTM-based models utilised by [15]'s research.
- In the decoding phase, we evaluated two greedy techniques to decode the predicted output of the acoustic model into its corresponding Gujarati text form.
- As a post-processing technique, we've adopted an enhanced spelling correction procedure. In other words, in addition to the insertion, deletion, replacement, and transposition of words, we have added a novel word-splitting feature to the algorithm. In addition, we include bi-gram terms alongside unigrams in the Guajarati corpus in order to rectify two words together that result in a decrease in WER.
- The proposed work decreased WER from 61.67 for greedy technique with MFCC and 49.48% for greedy technique with MFCC and CQCC to 44.21%.

3. Proposed Work

The proposed E2E voice recognition system is comprised of four main phases: Audio feature extractor, Deep speech architecture, decoding, and post processing shown in Figure 1. In the initial phase, the recorded audio speech/signal is processed and the relevant features are extracted by combining MFCC and CQCC. In the third phase, the output of the acoustic model is decoded using techniques such as greedy decoding. Post-processing is an additional phase added to the proposed E2E ASR system, which involves combining the improved Spell Corrector technique with the BERT model. After post-processing, the final predicted output text is the improvised output text. The following section elaborates on these main phases.

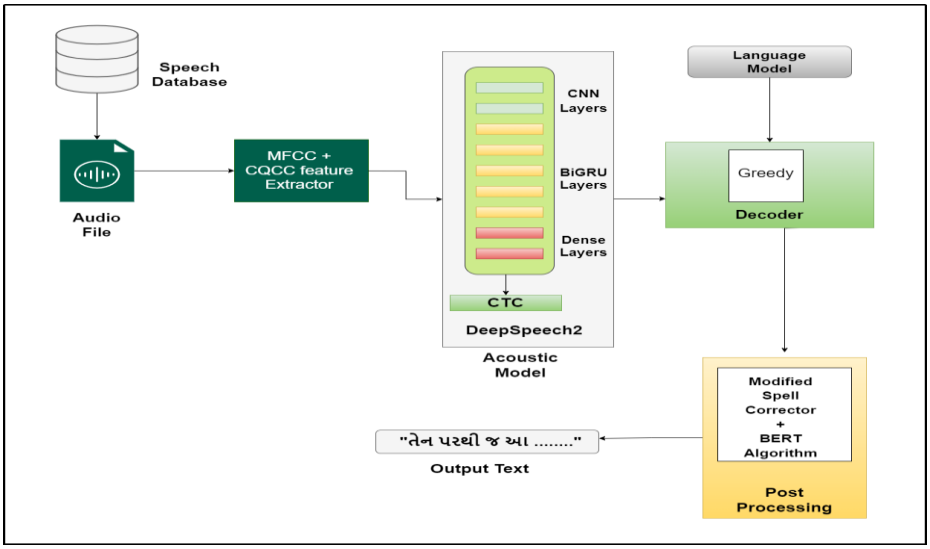

Fig. 1. Proposed End-to-End ASR Architecture

3.1 Audio Feature Extraction

MFCCs are commonly used for speech analysis and recognition due to their ability to capture or find the spectral characteristics of the human voice. CQCC, on the other hand, is a relatively new feature extraction technique that has been developed to capture the spectral information of audio signals more efficiently. Unlike MFCC, CQCC uses a constant-Q filter bank to obtain the spectral features. Combining MFCC and CQCC can result in a feature set that captures both the temporal and spectral characteristics of the audio signal. This can be beneficial in different applications such as speaker recognition, where the combination of the two feature sets has been shown to improve recognition accuracy.

We have used the combined MFCC and CQCC features to describe the input sound wave. These features have the batch size, time steps, and feature dimensions. These qualities are fed into the deep learning model. Our system recognises a unique utterance as $x_i$and a label as y_i both from the training set such as $X = (x_1, y_1), (x_2, y_2), \cdots\cdots\cdots\cdots$ etc. respectively. In this scenario, $y_i$is the produced output for the input signal x_i which has length$L$ with$y_s^i (0 \leq s \leq L-1)$is a only grapheme. The input signal x_i is a time series feature of length T, with x_i being the vector of STFT coefficients at time t $(0 \leq t \leq T-1)$.

3.2 DeepSpeech2 Model

As depicted in Figure 2, the CNN layers, batch normalisation layers, bidirectional GRU layers, dropout layers, dense layers, and CTC are the primary elements of the Deepspeech2-based acoustic model architecture for the proposed system. Depending on the amount of input data being processed, components such as CNN layers, bidirectional GRU layers, and dense layers are modified.

Frequency and time domain convolutions can be used to enhance ASR performance by modulating spectrum input properties [11]. Another argument for utilising convolution in the first layer [12] is that convolution in frequency attempt to replicate spectrum fluctuation caused by speaker variability. This is done by employing a ReLU activation function with a kernel size and a stride value of [11, 41] and two 2D-convolution layers containing 32 filters. Joint features, batch size, and time steps are the dimensions of the input features, while convolved time steps, filters, and batch size are the dimensions of the output features. The batch normalisation layer normalises the retrieved features before

sending them to the subsequent layer. Modern state-of-the-art recognizers typically employ Long Short Term Memory (LSTM) RNNs, which combine well with convolutional layers for feature extraction [13, 14].

Here, five layers of bidirectional GRU are employed, with each layer consisting of 128 bidirectional GRU units (256 GRU units). The bidirectional GRU layers generate outputs in response to inputs from the convolution layers (batch sample size, convolved time steps, and GRU units). In order to prevent over fitting, a 50% random exclusion was conducted. The extracted features are then supplied to the subsequent DNN-based layer. Given the input data, the output is (batch sample size, convolved time steps, and GRU units). During training, a standard technique for transforming audio input with variable length into output with variable length uses the CTC algorithm in combination with an RNN [15, 16, 17].

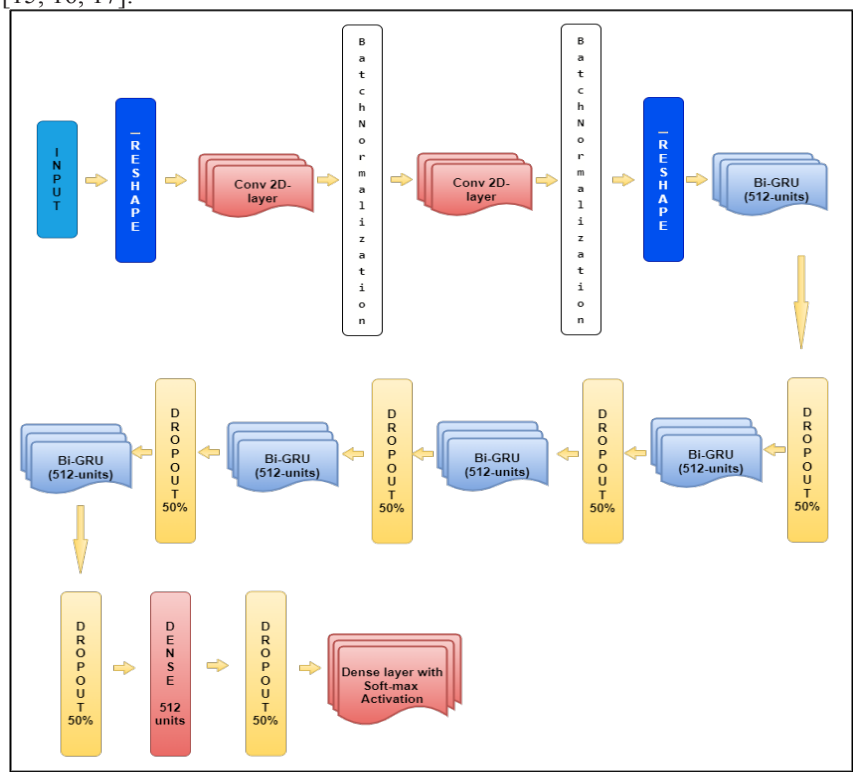

Fig. 2. Architecture of an acoustic model based on DeepSpeech2

3.3 Decoding

In ASR, the decoding technique plays a critical role in achieving accurate transcription of speech signals. One of the most frequently utilised decoding techniques of ASR systems is the greedy decoding technique. The greedy decoding technique is a type of beam search algorithm used in ASR systems for decoding speech signals. The algorithm works by keeping track of a set of candidate transcriptions, called the beam, at each decoding step. The algorithm selects the best candidates in the beam and continues to expand them until a transcription with the highest probability is found.

These decoding techniques have been applied in the implementation of the proposed ASR system. The greedy decoding techniques as a first approximation technique and constructed a character string l as stated in equation (1) by selecting the single most likely output at each timestamp (1). By combining similar characters and eliminating blank spaces, the following statement is translated into a transcript.

$$l = (s_1, s_2, s_3, \dots\dots\dots, s_t)\ where\ s_t = \underset{c \in \sum}{\operatorname{argmax}}\, p(c/x_t) \quad (1)$$

Since, it simply concatenates the outputs that are the most active at each time step, this initial estimate cannot be relied upon to identify the labelling that is most likely.

3.4 Post-Processing

The proposed method employs a BERT-based spell corrector as a post-processing technique with some enhancements to better the output of a voice recognition system. In addition to insertion, deletion, replacement, and transposition, we added a new mechanism for splitter words (Split) to the orthography corrector algorithm. In addition, we include bi-gram words in the Gujarati corpus to correct two words that have been combined, which further reduces WER. This enhanced orthography checker generates a list of substituted words that includes bigrams. Using the BERT model, the projected output words' orthography is adjusted. This method is referred to as BERT-based spell correction enhancement. We iterate over each word in a phrase, replacing each with its zero matching, one matching, or two matching Gujarati edit words from our corpus. If the anticipated phrase is correct, we can confirm using the list created by this approach that it already exists and does not need to be changed.

The list of sentences is then constructed by replacing each [MASK] term with each replacement term from the generated list of replacements, beginning with the starting word if it is not included in the generated list of replacements. The tokenized sentences are then sent to the BERT model. BERT generates a list of potential replacement words and their probabilities in relation to the sentence. The word substitutions with the highest probability from this list are added to the output list. Once every words have been iterated, a final output list containing all words with a maximal number of replacements for each word is generated.

4. Experiment Setup and Results

By modifying particular parameters and hyper-parameters based on previous research in the low-resource Gujarati language and the high-resource English language, we customised the system for the low-resource Gujarati language [6, 25]. Previous efforts in the limited-resource gujarati language [15] inspired our proposed effort. We used the crowd sourced high quality gujarati language multi speaker speech dataset to implement the proposed task. There are two categories of speakers in the sample for which Multi-Speaker ASR system models are proposed in this paper. We configured a system with the Windows 10 operating system, 8GB RAM, and a 12GB RAM Tesla T4 GPU. The model was trained for 35 epochs over 26.5 hours using 15 collections of training and validation data for multi-speaker models. There are a total of 26,675,630 parameters in the system. The Adam optimizer was utilised to perform gradient descent, and the CTC loss function was utilised to calculate the loss. Losses calculated per epoch are depicted in Figure 3.

4.1 Performance Analysis of Multi-speaker ASR System

Multi-Speaker ASR system was developed by using 3,844 utterances of multi-speaker dataset out of total 4,272 utterances (25 hours) for training, 214 utterances (1.5 hours) for validating the model, and the remaining 214 utterances (1.5 hours) for testing the model. Table 1 reveals that the WER of the multi-speaker ASR system has decreased by 17.46% compared to the initial WER of 61.67 for Greedy decoding with MFCC.

Table 1. Performance Analysis of Multi-speaker ASR System

System	WER (%)
MFCC + Multi-speaker ASR Model + Greedy Decoding	61.67

MFCC+CQCC + Multi-speaker ASR Model + Greedy Decoding	49.48
MFCC+CQCC + Multi-speaker ASR Model + Greedy Decoding & Modified Spell Corrector	44.21

4.2 Loss graph for Multi-speaker System with MFCC+CQCC and only MFCC

This section describes the training and validation loss value for ASR systems with greedy decoding and multiple speakers ASR model. Figure 3 demonstrates that the loss value is lower for multi-speaker ASR systems with integrated MFCC+CQCC feature compared to single MFCC features. In this paper, we conclude that multi-speaker ASR and greedy decoding with MFCC+CQCC has lower WER and loss function than only MFCC multi speaker system.

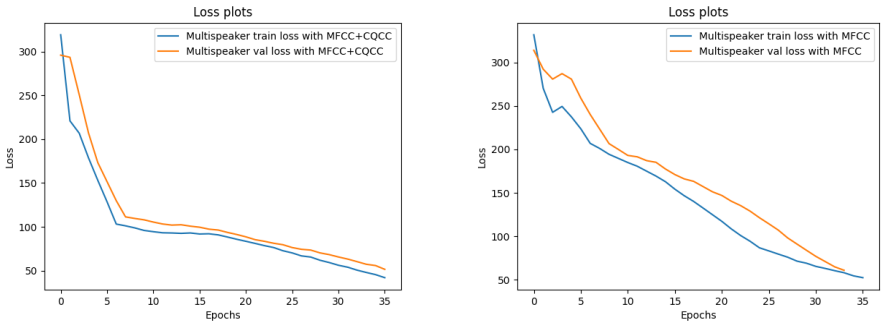

Fig. 3. System loss value plot

4.3 Comparing with Existing Works

For improving the efficacy of E2E gujarati language ASR systems, the MFCC+CQCC integrated feature extraction technique provides a superior method for extracting gujarati speech features from audio signals. The proposed work is an extension of some previously proposed state-of-the-art works that used MFCC, Log frequency spectrogram, VTLN factor, and frame rate features with CNN-BiLSTM, RNN-CTC, Monolingual or Multilingual models. Table 2 summarises the outcomes of the proposed ASR system, which combines greedy decoding and post-processing techniques with the DeepSpeech2 model at the back end and MFCC+CQCC feature extraction types at the front end. According to our testing, MFCC+CQCC features with DeepSpeech2 architecture and modified spell corrector based BERT model as a post-processing methods outperform previous combination result sets. We can observe that the proposed work lowers the WER by 17.46% compared to the existing method, which reduces WER by 5.86%. Comparing the outcomes of our method with those of other models, table 2 reveals that MFCC+CQCC features with DeepSpeech2 architecture and greedy decoding method with the Modified Spell Corrector BERT algorithm performed better than other models.

Table 2. Comparison with Existing Works

Work	**Feature Extraction**	**Classifier**	**Decoding + PostProcessing**	**Dataset**	**Parameter**	
					WER (%)	**Loss**
Amodei et al. [9]	Log – Frequency Spectrogram	Uni-directional RNN-CTC, GRU-	Wide beam search	WSJ eval'93, LibriSpeech test-other, VoxForge Indian,	4.42, 12.73, 22.89, 21.59	NA

		CTC		CHiMEeval real (English)		
Billa [25]	VTLN factor and Feature frame rate	LSTM-CTC, Monolingual or Multilingual Training	NA	CMU-INDIC dataset (Gujarati)	20.91, 21.44, 19.33, 19.30	NA
Raval et al. [33]	MFCC	CNN, BiLSTM, CTC	Greedy, Prefix with LMs', Prefix with LMs'and Spell corrector BERT	Microsoft Speech Corpus (Gujarati)	70.65, 69.94, 64.78	NA
Proposed Approach	MFCC+ CQCC Integrated features	CNN, Bidirectional GRU, CTC, Multi-Speaker Model	Greedy decoding and Modified spell corrector BERT	Crowdsourced, high-quality, and multi-speaker (male and female) Gujarati language dataset	61.67, 49.48, 44.21	45.2, 56.5

5. Conclusion

In this study, an E2E gujarati language based ASR system is developed. The proposed work improve the performance of the system, by using MFCC+CQCC integrated feature extraction technique with Deepspeech2 model and greedy decoding, and during the post processing phases, we applied a BERT based spelling corrector model with an additional split feature. We find that the proposed strategy lowers the WER overall by 17.46%.

Although effective deep learning models require huge amounts of training data, this article demonstrates that performance can be improved without adding more training data. This is crucial for a language like Gujarati with limited resources. We are confident that our enhancements would perform significantly better if we had more data.

References

[1] Baker, James. "The DRAGON system--An overview." IEEE Transactions on Acoustics, speech, and signal Processing 23.1 (1975): 24-29.

[2] Deshmukh, Akshay Madhav. "Comparison of hidden markov model and recurrent neural network in automatic speech recognition." European Journal of Engineering and Technology Research 5.8 (2020): 958-965.

[3] Forsberg, Markus. "Why is speech recognition difficult." Chalmers University of Technology (2003).

[4] Dua, Mohit. "Gujarati Language Automatic Speech Recognition Using Integrated Feature Extraction and Hybrid Acoustic Model." Proceedings of Fourth International Conference on Communication, Computing and Electronics Systems:

ICCCES 2022. Singapore: Springer Nature Singapore, 2023.

[5] Chakravarty, Nidhi, and Mohit Dua. "Spoof Detection using Sequentially Integrated Image and Audio Features." International Journal of Computing and Digital Systems 13.1 (2023): 1-1.

[6] Amodei, Dario, et al. "Deep speech 2: End-to-end speech recognition in english and mandarin." International conference on machine learning. PMLR, 2016.

[7] Graves, Alex, et al. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks." Proceedings of the 23rd international conference on Machine learning. 2006.

[8] Bourlard, Herve A., and Nelson Morgan. Connectionist speech recognition: a hybrid approach. Vol. 247. Springer Science & Business Media, 1994.

[9] Raval, Deepang, et al. "Improving Deep Learning based Automatic Speech Recognition for Gujarati." Transactions on Asian and Low-Resource Language Information Processing 21.3 (2021): 1-18.

[10] Zhang, Shaohua, et al. "Spelling error correction with soft-masked BERT." arXiv preprint arXiv:2005.07421 (2020).

[11] Toshniwal, Shubham, et al. "Multilingual speech recognition with a single end-to-end model." 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2018.

[12] Billa, Jayadev. "ISI ASR System for the Low Resource Speech Recognition Challenge for Indian Languages." INTERSPEECH. 2018.

[13] Sak, Haşim, Andrew Senior, and Françoise Beaufays. "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition." arXiv preprint arXiv:1402.1128 (2014).

[14] Schuster, M., and K. K. Paliwal. "Networks bidirectional reccurent neural." IEEE Trans Signal Proces 45 (1997): 2673-2681.

[15] Graves, Alex, and Navdeep Jaitly. "Towards end-to-end speech recognition with recurrent neural networks." International conference on machine learning. PMLR, 2014.

[16] Hannun, Awni, et al. "Deep speech: Scaling up end-to-end speech recognition." arXiv preprint arXiv:1412.5567 (2014).

[17] Maas, Andrew, et al. "Lexicon-free conversational speech recognition with neural networks." Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015.