INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Low Resource Kannada Speech Recognition using Lattice Rescoring and Speech Synthesis

Savitha Murthy[1]*, **Dinkar Sitaram**[2]

1 PhD Scholar, Department of CSE, PES University, India
2 Director, Cloud Computing Innovation Council of India, India

OPEN ACCESS

Received: 10-12-2022

Accepted: 05-01-2023

Published: 31-01-2023

Citation: Murthy S, Sitaram D (2023) Low Resource Kannada Speech Recognition using Lattice Rescoring and Speech Synthesis. Indian Journal of Science and Technology 16(4): 282-291. https://doi.org/10.17485/IJST/v16i4.2371

* **Corresponding author**.

savithamurthy@pes.edu

Funding: None

Competing Interests: None

Copyright: © 2023 Murthy & Sitaram. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: Improving the accuracy of low resource speech recognition in a model trained on only 4 hours of transcribed continuous speech in Kannada language, using data augmentation. **Methods:** Baseline language model is augmented with unigram counts of words, that are present in the Wikipedia text corpus but absent in the baseline, for initial decoding. Lattice rescoring is then applied using the language model augmented with Wikipedia text. Speech synthesis-based augmentation with multi-speaker syllable-based synthesis, using voices in Kannada and cross-lingual Telugu languages, is employed. We synthesize basic syllables, syllables with consonant conjuncts, and words that contain syllables that are absent in the training speech, for Kannada language. **Findings:** An overall word error rate (WER) of 9.04% is achieved over a baseline WER of 40.93%. Language model augmentation and lattice rescoring gives an absolute improvement of 16.68%. Applying our method of syllable-based speech synthesis over language model augmentation and rescoring yields a total reduction of 31.89% in WER. The proposed approach of language model augmentation is memory efficient and consumes only 1/8th the memory required for decoding with Wikipedia augmented language model (2 gigabytes versus 18 gigabytes) while giving comparable WER (22.95% for Wikipedia versus 24.25% for our method). Augmentation with synthesized syllables enhances the ability of the speech recognition model to recognize basic sounds thus improving recognition of out-of-vocabulary words to 90% and in-vocabulary words to 97%. **Novelty**: We propose novel methods of language model augmentation and synthesis-based augmentation to achieve low WER for a speech recognition model trained on only 4 hours of continuous speech. Obtaining high recognition accuracy (or low WER) for very small speech corpus is a challenge. In this paper, we demonstrate that high accuracy can be achieved using data augmentation for a small corpus-based speech recognition.

Keywords: Low resource; Speech synthesis; Data augmentation; Language model; Lattice rescoring

1 Introduction

Human-computer interaction (HCI) is an important part of Artificial Intelligent (AI) systems. Automatic Speech Recognition (ASR) is the gateway to HCI making it an inherent part of artificial intelligence systems. ASR enables computers to process and recognize human speech, thus enabling HCI. Training an efficient ASR system requires large amounts of transcribed speech data of the order of hundreds of hours. Only a few languages in the world, notably English, have sufficient speech corpora. Other languages that have less than 100 hours of transcribed speech data are classified as low resource languages[(1)].

In this paper, we focus on ASR for a low resource language, namely Kannada, an Indic language.

India is a country with 22 official languages. Additionally, around 122 major languages are spoken in India[(2)]. Developing speech recognition applications for all the Indic languages is the ask of the hour. Most of these languages are low resource with a scarcity of sufficient labelled audio[(3)]. As per the survey conducted by[(4)], ASR has been implemented for a very few Indic languages such as Hindi, Tamil, Marathi, Punjabi, Gujarati, Telugu, Kannada, Bengali, Manipuri, Assamese, and Odia languages. There have been several efforts to develop an ASR for Kannada language. The authors in[(5)] summarize the ASR studies in Kannada. The works in[(6,7)] focus on recognizing spoken isolated words in Kannada. ASR for continuous Kannada speech has been developed by the studies in[(8–10)]. Achieving an accuracy of 90% or more for a continuous ASR requires sufficient amount of training data. Recent work in Kannada speech recognition is by Kumar and Jayanna[(8)] who train an ASR model on approximately 25 hours of continuous speech Kannada dataset and obtain a low word error rate (WER) of 4.05%. The work in[(9)] also involves ASR development on a similar corpus but under noisy conditions. Collecting a corpus having several hours of speech data together with corresponding transcripts (labeled speech) is a tedious task. The objective of our work in this paper is to achieve a small WER, which is less than 10%, for ASR trained on only 4 hours of continuous labeled speech in Kannada. Based on the detailed literature survey conducted by us, such accuracies have not been achieved for ASR trained on a very small continuous speech corpus.

We apply data augmentation to achieve our objective. Data augmentation is an approach to improve the recognition accuracy of ASR without having to collect additional speech data. The work by Chellapriyadarshini et al. in[(11)] applies data augmentation to a small speech corpus of 3.74 hours in Tamil. They apply semi-supervised training as a data augmentation technique and achieve an absolute improvement of 2.8%, over the baseline, resulting in a WER of 27.6%. In semi-supervised training, baseline ASR is enhanced with additional unlabeled speech that is transcribed and labeled using the baseline ASR itself. However, their work achieves a small improvement because the WER of the baseline is 30.4% which is high for semi-supervised learning to be efficient.

Language model augmentation and speech synthesis-based augmentation are two other data augmentation techniques that have been applied for Kannada language in[(12)]. Language model augmentation involves enhancing the language model with text from an external text corpus, for example, Wikipedia. Speech synthesis-based augmentation is a technique where the training speech data is enhanced with synthesized speech to improve the acoustic capability of the ASR. The work by Murthy et al. in[(12)] applies language model augmentation and synthesis-based augmentation on a low resource continuous speech dataset of 4 hours in Kannada. The authors achieve an absolute improvement of 8.62% over the baseline, resulting in WER of 38.02%. This means more than 30% of words are misrecognized during decoding which may be unacceptable in real-life ASR.

In this paper, we propose novel methods of language model augmentation and syllable-based speech synthesis to improve speech recognition accuracy for Kannada ASR and achieve WER as low as 9.04%, starting from a baseline WER of 40.93%, using 4 hours of continuous speech dataset in Kannada. In particular, our method of language model augmentation gives an absolute improvement of 16.68% over the baseline. On further applying our method of syllable-based speech synthesis, we achieve a total improvement of 31.89% over the baseline. Additionally, our method of language model augmentation consumes only 1/8th the memory required to decode with a large language model while giving comparable accuracy.

Our research contributions are as follows:

1. Enhance acoustic learning of the ASR with synthesized sounds of syllables that form the basic independent pronunciation units of the Kannada language.

2. Enable improved learning of pronunciation contexts by integrating synthesized conjunct consonants and synthesized words that contain syllables absent in the ASR training data.

3. Memory efficient decoding, using a baseline language model augmented with only out-of-vocabulary (OOV) words, to achieve improved lattice rescoring with a larger language model (OOV words in this context are those words that are present in Wikipedia but not in the baseline text).

The objective of our work is to motivate ASR development for other Indic languages which lack sufficient amount of transcribed speech.

2 Methodology

We employ hybrid ASR architecture for our study on Kannada language speech recognition. In section 2.1, we describe the concept of hybrid ASR including ASR decoding, lattice rescoring, performance evaluation, and the ASR implementation in this paper. Section 2.2 outlines the speech corpus used for our experiments. In section 2.3, we describe the two components of our proposed approach, namely, language model augmentation for lattice rescoring and augmentation using syllable-based speech synthesis.

2.1 Hybrid ASR

Conventional ASR architectures comprise of a pronunciation lexicon, language model, and acoustic model. The pronunciation lexicon models the pronunciations of words in the vocabulary as a phoneme sequence. The language model defines the word sequence probabilities and is trained on a text corpus of the target language. The acoustic model represents the probabilities of pronunciations and is trained on speech-text pairs in the target language with acoustic features extracted from the audio for training. The objective of an ASR is to maximize the probabilities of the language model, lexical model and acoustic model during inference as given in Equation (1).

$$W^{*} = \text{argmax}_{w} \Sigma P(O \mid Q) P(Q \mid W) P(W) \tag{1}$$

where, W^{*} is the hypothesis, which is a sequence of words inferred; $P(W)$ represents the language model probabilities, and W is the word sequence; $P(Q|W)$ represents the lexical model probabilities, and Q is the state sequence; and $-$ represents the acoustic model probabilities, and O is the sequence of acoustic observations. For acoustic modeling, GMMs were earlier used to model the pronunciations of phonemes which form the basic sounds, and Hidden Markov models (HMM) to model the phoneme sequences. Deep Neural Networks (DNN) were later employed to model pronunciations because of their efficiency and robustness in modelling nonlinear data. The DNN-HMMs then replaced GMM-HMMs for acoustic modeling which were called "hybrid ASR architectures".

2.1.1 ASR Decoding

The hypothesis of an ASR is obtained after decoding the input speech. Decoding involves finding the most probable path of word sequences. A Weighted Finite State Transducer (WFST)[13] is used for this purpose. It is represented as given in Equation (2).

$$HCLG = \min(\det(H \circ C \circ L \circ G)) \tag{2}$$

where H represents the HMM transitions, C represents context dependencies among phonemes, L represents the lexicon and G represents the grammar represented by the language model.

While decoding with a bigger language model may be more effective in reducing WER, composing a HCLG graph, also known as decoding graph, with the grammar belonging to a larger language model is more memory intensive and decoding becomes computationally expensive because of a larger search space. Hence, normal practice is to decode using the smaller language model and then perform lattice rescoring with a larger language model.

2.1.2 Lattice Rescoring

Lattice represents alternate possible word sequences that have higher scores than other possible word sequences. Decoding generates lattices. A lattice contains the paths with higher scores and is derived from the pruned subset of the decoding graph[13]. Lattice rescoring is a process where the path probabilities are updated with the new language model probabilities while retaining the transition and pronunciation probabilities. Lattice rescoring can be performed using large language models for improved results. This also conserves computational resources as there is no need to compose a decoding graph.

Performance Evaluation of ASR

The performance of an ASR is measured in terms of WER which is given by equation (2).

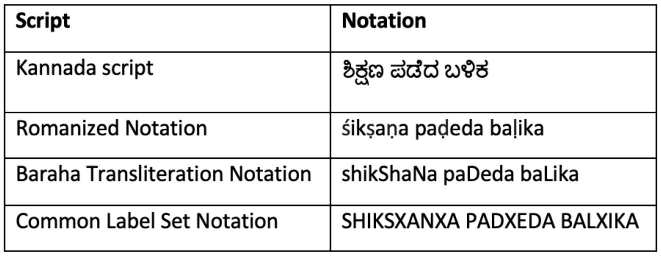
$$WER = \frac{I + D + S}{N} \tag{2}$$

where 'I' denotes the number of insertions, 'D' denotes the number of deletions, 'S' denotes the number of substitutions and 'N' represents the total number of words in the ASR hypothesis.

2.1.4 ASR Implementation

We implement a hybrid ASR framework for our research. The ASR model is trained using Kaldi toolkit. We employ a DNN-HMM setup for our experiments on Kannada ASR. The recipe for training the DNN model is Karel's DNN setup. The parameters used are the same as the baseline parameters specified in Interspeech Microsoft Challenge, 2018 [3]. The baseline DNN consists of 7 hidden layers with a dimension of 2048, an initial learning rate of 0.008 and an acoustic weight of 0.08.

The language model for the baseline ASR is trained on the speech transcripts of the baseline training dataset. SRILM toolkit is employed to train the language model. Trigram language model with Witten bell smoothing is used. Lexicon is defined using word-to-phoneme mappings based on the orthographic rules of Kannada. A phone set of 48 phonemes including silence is defined. The lexicon is built from the prepared common label set notation, provided by IIT-Madras (https://www.iitm.ac.in/donlab/tts), using the transcripts of the training set as depicted in Figure 1.



Script	Notation
Kannada script	ಶಿಕ್ಷಣ ಪಡೆದ ಬಳಿಕ
Romanized Notation	śikṣaṇa paḍeda baḷika
Baraha Transliteration Notation	shikShaNa paDeda baLika
Common Label Set Notation	SHIKSXANXA PADXEDA BALXIKA

Fig 1. Example of different Kannada notations

2.2 Speech Corpus

We developed a small Kannada continuous speech corpus consisting of 46 speakers and 2647 utterances. The Kannada transcripts for recording were borrowed from speech synthesis dataset defined by IIIT-Hyderabad (http://festvox.org/databases/iiit_voices). The transcripts contain 1000 sentences in Kannada with a vocabulary of 1754 words. The training dataset consists of 34 speakers with 4 hours of speech. The test dataset consists of 1.64 hours of speech from 12 speakers. We ensure that there is no overlap between the utterances (spoken sentences) and speakers of the train and the test set to simulate real-life conditions.

The speech synthesis transcripts contained characters other than letters such as punctuation. The transcripts were cleaned to remove punctuation characters and then transliterated to English alphabets using Baraha software. The Baraha English scripts were then converted to notations from the common label set. Figure 1 depicts an example of different notations for Kannada including the IIT-M label set notation.

2.3 Data Augmentation

The proposed approach in this paper is depicted in Figure 2. Our approach contains two components – the first component is to augment the baseline language model with OOV words for initial decoding and then perform lattice rescoring with a large language model augmented using Wikipedia text as described in section 2.3.1, and the second component is speech synthesis based augmentation using synthesized syllable-based sounds and words as explained in section 2.3.2.

2.3.1 Language model augmentation and lattice rescoring

Studies that involve language model augmentation select sentences from a large external text corpus based on certain scores assigned to the sentences [12,14]. There is always a question of how much to select without making the augmented language model size very large for decoding. For example, the work in [12] selects the first 50 sentences of Kannada Wikipedia that contain certain OOV words. However, in case of only 4 hours of baseline speech, every sentence in a large corpus may contain an OOV word. Hence, in order to leverage the availability of an external text corpus (such as Wikipedia) to the full extent, we propose a novel approach of augmenting the baseline language model with only OOV words for initial decoding. This method does not make the language model very large, at the same time, the lattices generated are comprehensive enough for effective lattice rescoring with a larger language model (such as Wikipedia).

We implement language model augmentation by interpolating a new language model and the baseline language model. The "count merging" method [15] of interpolation is employed to merge the new language model with the baseline language model.

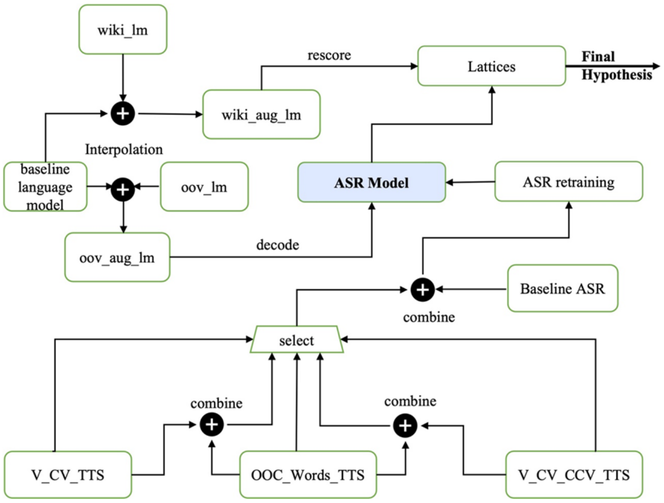


Fig 2. Proposed Approach

Count merging is represented by equation (3).

$$p^{CM}(w|h) = \frac{\sum_i \beta_i c_i(h)\, p_i(w|h)}{\sum \beta_j c_j(h)} \tag{3}$$

where 'i' represents the domain or the model, 'β_i' is the model scaling factor for 'i', and 'h' is the history for the i^{th} model or domain. The interpolation weight for a given history in count merging is given as stated in equation (4).

$$\lambda_i(h) = \frac{\beta_i c_i(h)}{\sum_j \sum \beta_j c_j(h)} \tag{4}$$

The interpolation weight in count merging depends on the history counts pertaining to a particular domain instead of absolute counts. Thus, the history counts that belong to a smaller corpus get more weightage as compared to the same history in a larger corpus which is suitable for low resource language.

We develop two new augmented language models using interpolation, namely, "wiki_aug_lm" and "oov_aug_lm", described as follows.

wiki_aug_lm: We train a language model using normalized Wikipedia text, called "wiki_lm". Normalization involves eliminating punctuation and non-language characters and replacing numerals with text. After normalization, Kannada Wikipedia text consists of 873339 sentences increasing the vocabulary to 943729 words from an initial vocabulary of 1754 in the baseline. A new language model, "wiki_aug_lm", is obtained by interpolating the "wiki_lm" with the baseline language model for decoding.

oov_aug_lm: We train a language model only on the unigram counts of OOV words from the Wikipedia corpus, called "oov_lm", and interpolate it with the baseline language model to obtain a new language model, called "oov_aug_lm", augmented with only OOV words.

Decoding is performed with 'oov_aug_lm' to generate lattices. The lattices are then rescored with a larger 'wiki_aug_lm' shown in Figure 2.

2.3.2 Data augmentation using syllable-based speech synthesis

ASR for a low resource language trained on only 4 hours of transcribed speech, may not contain all the possible sound sequences. In other words, all possible contexts for every phoneme may not be present in the training data. For example, the total number of syllables (syllables comprise one or more phoneme sequences) largely used in Kannada can be estimated to be approximately 15000 or more. In such a case, along with text augmentation, enhancing the acoustic capability of the ASR using speech synthesis may be useful to improve accuracies.

The work in [12] employs speech synthesis to address OOV detection and recovery by synthesizing only certain OOV words and syllables. However, synthesizing all the OOV words, for a very small corpus containing only 4 hours of natural speech, would result in a large amount of synthesized audio thus overwhelming the ASR leading to high WERs. We propose a novel approach of including only synthesized syllables and certain synthesized words that contain missing syllables as an acoustic data augmentation technique for the hybrid ASR model. Our approach is more comprehensive with inclusion of all sounds in Kannada language while keeping the amount of synthesized speech low.

We retrain the ASR after integrating the baseline speech data with synthesized syllables and synthesized words. The syllable sounds are generated using Indic voices using Festival Text-to-Speech (TTS) system. Labeled speech from other languages has been employed in literature to leverage the similarity of speech (phonetics), in the context of Indian languages, using a technique known as multilingual training[(16)]. Our method of syllable-based speech synthesis uses multilingual voices for acoustic augmentation by employing both Kannada and Telugu voices. Voices from Telugu are chosen because the alphabets and phonetics of Telugu are very similar to that of Kannada. Three Telugu language voices along with one Kannada voice available with the Festival TTS are used for multi-speaker synthesis of syllables and words. Kannada voice is from a female speaker while Telugu voices include one male speaker and two female speakers. The length of each synthesized audio is about 1 to 2 seconds.

We apply three different sets of synthesized audios for retraining the ASR, generating three different ASR models as depicted in Figure 2. The synthesized datasets and corresponding ASR models are as follows – (i) inclusion of synthesized basic syllables of the Kannada language, called "Model V_CV_TTS", (ii) inclusion of synthesized sounds of consonant conjuncts along with basic syllables, called "Model V_CV_CCV_TTS", and (iii) inclusion of synthesized OOV words that contain out-of-corpus (OOC) syllables, called "Model OOC_Words_TTS". The details of the models are given as follows:

Model V_CV_TTS: The baseline speech data is combined with synthesized sounds of V (vowels) and CV (consonant-vowel) sequences for retraining the ASR. Here, we synthesize only the basic alphabet set of the Kannada language which consists of 15 vowels known as 'swaras' and 34 consonants known as 'vyanjanas'. Theoretically, this results in 15 x 34 = 510 CV sequences. This set comprises the context-independent pronunciation units in Kannada. We consider 14 vowels (omitting visarga ':') and 447 out of 476 CV sequences that are used in spoken Kannada for synthesis. The ones that are not frequently used, for example, the sequences comprising the first two 'anunasikas' (nasal consonant) are omitted.

Model V_CV_CCV_TTS: The baseline speech data is combined with synthesized sounds of V, CV and CCV sequences for retraining the ASR. Here, along with V and CV sequences, we also synthesize CCV sequences. Synthesizing CCV accommodates all possible contexts for the conjunct consonants known as 'vattaksharas'. Theoretically, 15 vowels and 34 consonants would result in 14 x 34 x 34 = 17,340 CCV sequences and more than half a million if CCCV forms are included. For practical purposes, we consider 14335 CCV sequences along with 14 vowels and 447 CV sequences resulting in 14872 different pronunciations for synthesis. CCCV forms are very rare and are not included in our study.

Model OOC_Words_TTS: We identify those syllables of the Kannada language that are present in the Wikipedia text but absent in the training corpus. Such syllables are called OOC syllables and the words in which these syllables occur as OOC words. Words from Wikipedia containing OOC syllables are identified for synthesis. We ensure that the three possible contexts for the OOC syllables are considered by selecting words that contain these syllables at the beginning, in the middle, and at the end. The baseline speech data is combined with synthesized audio of words containing OOC syllables for retraining the ASR. Considering all positions of syllables for word selection ensures coverage of different possible contexts in the presence of conjuncts.

Combination of synthesis models: We also evaluate the effect of combining different sets of synthetic data on WER, namely, "V_CV_TTS + OOC_Words_TTS", and "V_CV_TTS + CCV_TTS + OOC_Words_TTS" models.

3 Results and Discussion

We discuss the results obtained by our experiments in this section. Section 3.1 discusses the effect of our proposed language model and rescoring technique. The results of integrating syllable-based speech synthesis into the ASR are discussed in section 3.2. Section 3.3 outlines the effect of our approach on OOV and in-vocabulary (IV) words. We compare our work with previous works in section 3.4.

3.1 Language Model Augmentation and Rescoring

Decoding with a very large language model is not always practical because the memory requirement for decoding graph construction with such models can be very high. For example, a language model augmented with full Wikipedia text, namely "wiki_aug_lm", for Kannada requires 18 gigabytes (GB) of memory as listed in Table 1. This makes it difficult to run decoding in parallel leading to more decoding time. Therefore, we compare only OOV based enhancements to the baseline language model and their memory requirements as depicted in Table 1.

Table 1. Memory requirements for decoding graph construction

Language Model	Maximum Memory Required
baseline	~ 1 GB
wiki_aug_lm	~ 18 GB
baseline + Wikipedia OOV lines	~ 18 GB
baseline + Wikipedia OOV trigrams	~26 GB
oov_aug_lm	< 2 GB

GB: GigaBytes

We consider three kinds of OOV based language model augmentations using Wikipedia Kannada language text. First, the Wikipedia lines containing OOV words are selected for enhancing the baseline language model. In the case of low resourced, agglutinative, and inflective languages like Kannada, lines containing OOV constitute more than 90% of Wikipedia text. Augmenting the baseline language model with such a big subset again requires approximately the same memory as full Wikipedia augmentation. Second, the trigrams from Wikipedia containing OOV words are selected. Again because of high OOV rates, this results in a bigger language model than the full Wikipedia itself. As specified in Table 1, "baseline + Wikipedia OOV trigrams" requires more memory than "wiki_aug_lm". Third, the baseline language model is augmented with unigram counts of only OOV words, namely "oov_aug_lm". As seen in Table 2, "oov_aug_lm" requires 2 GB memory which is only 1/8th the memory needed to decode using "wiki_aug_lm". Therefore, we employ initial decoding with a language model that is augmented with OOV unigram counts from Wikipedia, i.e., "oov_aug_lm".

For comparative analysis, we perform two separate lattice rescoring. First, we decode with the baseline language model and rescore the generated lattice with "wiki_aug_lm". Second, we decode with OOV augmented language model i.e., "oov_aug_lm", and rescore the generated lattice with "wiki_aug_lm". We compare the rescoring results with the best accuracy obtained from full decoding using "wiki_aug_lm". The results are listed in Table 2.

Table 2. Decoding and lattice rescoring results

Language Model	WER (%)		
	Decoding	Lattice Rescoring	Rescore after oov_aug_lm decode (our method)
Baseline	40.93	-	-
wiki_aug_lm	22.95	39.34	24.25

As depicted in Table 2, rescoring the lattices obtain after decoding with the language model augmented with only OOV words results in a WER of 24.25%, which is only less 1.3% more than the WER obtained by decoding with "wiki_aug_lm".

3.2 Inclusion of synthesized audio into the acoustic model

The results of integrating the baseline ASR with different synthesized datasets are listed in Table 3. WERs for evaluation are obtained by decoding the test set using "oov_aug_lm" and rescoring with "wiki_aug_lm" as specified before.

Table 3. Results of integrating different synthesized datasets

ASR model	WER (%)
V_CV_TTS	13.34
V_CV_CCV_TTS	17.13
OOC_Words_TTS	16.22
V_CV_TTS + OOC_Words_TTS	9.04
V_CV_CCV_TTS + OOC_Words_TTS	16.46

The inclusion of V_CV_TTS shows an absolute improvement of 27.59% over the baseline and an absolute improvement of 10.91% over language model augmentation and rescoring. The inclusion of V_CV_CCV_TTS sounds shows 3.79% less improvement as compared to V_CV_TTS. This can be due to the following facts – first, the duration of the synthesized audio is very large in the case of V_CV_CCV_TTS, and second, CCV forms contain half consonant sounds at the beginning which are context dependent and are better recognized with context. The best absolute improvement of 31.89% over the baseline and an absolute improvement of 15.21% over language model augmentation and rescoring is obtained by including the combination of synthesized data comprising of V, CV, and OOC words. This results in the lowest WER of 9.04%. Also, the results show that using a combination of synthesized syllables and words for enhancing the acoustic data results in better accuracy than augmenting with synthesized OOV words alone.

Table 4. Hypotheses after integrating V_CV_TTS, V_CV_CCV_TTS, OOC_Words_TTS and combined datasets

ASR Model	Hypothesis – Example 1	Hypothesis – Example 2
Ground-truth	DHAARAWAADXA …	”... NAWA BRXQDAAWANA ...”
Baseline + wiki_lm	BHAANUWAARA **…**	”... NAWA BRIQDAABAN ...”
V_CV_TTS	DHAARAWAADXA	”... NAWA WRXQDAAWANA ...”
V_CV_CCV_TTS	II DHAARAWAADXA ….	”... NAWA BRXQDAAWANA ...”
OOC_Words_TTS	DHAARAWAADXA …	”... NAWA WRXQDAAWANA ...”
V_CV_TTS + OOC_Words_TTS	DHAARAWAADXA …	”... NAWA BRXQDAAWANA ...”
V_CV_CCV_TTS + OOC_Words_TTS	DHAARAWAADXA …	”... NAWA BRXQDAAWANA ...”

the example hypotheses obtained after augmenting with different synthesized datasets. The word “DHAARAWAADXA”, being an OOV, is not recognized correctly by wiki_lm and is recognized as “BHAANUWAARA” instead. V_CV_TTS augmentation recognizes the word correctly. However, the insertion of ‘II’ is present before “DHAARAWAADXA” in the case of V_CV_CCV_TTS augmentation in spite of the word being recognized correctly. This is due to over fitting. The models with OOC_Words_TTS, and the combinations of “V_CV_TTS + OOC_Words_TTS” and “V_CV_CCV_TTS + OOC_Words_TTS” recognize the word correctly without any insertions because the inclusion of synthesized words with contexts acts as a regularizer. Similarly, the syllable “BRXQ” is an OOC syllable and the word “BRXQDAAWANA” is an OOV word. Only text augmentation results in the word being recognized as “BRIQDAABAN”, “BRIQ” being the closest pronunciation distribution for “BRXQ”. The addition of synthetic audio with V_CV_CCV_TTS seems to help the recognition of the context “BRXQDAA” correctly, while V_CV_TTS and OOC_Words_TTS result in “BRXQ” being recognized as “WRXQ”, which is present in the training data, showing lack of reinforcement of the syllable sound. However, inclusion of a combination of synthesized datasets results in the correct recognition of ”BRXQDAAWANA”.

The duration of TTS audio for different models and word error rates are depicted in Figure 3. Best accuracy is reached with a duration of approximately 30 hours of TTS audio and a combination of synthesized syllables and words i.e. “V_CV_TTS + OOC_Words_TTS”. The combination model “V_CV_CCV_TTS + OOC_Words_TTS”, using one Kannada and three Telugu voices generates 174.89 hours of audio which is very large when compared to 4 hours of the original recording and hence results in less improvement. However, the WER for this model is lower than using only model “V_CV_CCV_TTS”. This demonstrates that a combination of both syllables and words used for synthesis is more effective in improving ASR accuracy in the case of very low resource ASR.

We verified the effect of reducing the duration of TTS audio, from 174.89 hours to 91.37 hours, by employing only one Telugu female voice and one Kannada voice to generate the combination model “V_CV_CCV_TTS + OOC_Words_TTS” of synthesized audio. Retraining the ASR with reduced synthesized audio of 91.37 hours reduced the WER to 11.62%.

3.3 Effect on recognition of OOV and IV words

The effect of language model augmentation and inclusion of synthesized audio on recognition of OOV words and IV words is depicted in Table 5. Rescoring with Wikipedia augmented language model, before inclusion of synthesis speech, facilitates the recognition of 62.28% of OOV. Improvement of recognition in both OOV words and IV words is obtained the inclusion of synthesized speech. There is a noticeable increase of 22.51% in OOV recognition over language model augmentation due to the inclusion of synthesized sounds of only basic syllables i.e., “V_CV_CCV_TTS”. This demonstrates that enhancing the acoustic capability of the ASR with basic sounds improves recognition. Also, the highest recognition of OOV words is achieved with inclusion of “V_CV_CCV_TTS + OOC_Words_TTS” at ~91% which is one per cent more than the inclusion of “V_CV_TTS + OOC_Words_TTS” which is ~90%. Recognition of IV words is also at its best with ~98% for “V_CV_CCV_TTS + OOC_Words_TTS” which is one percent more than that for “V_CV_TTS + OOC_Words_TTS” which is ~97%. Inclusion of only V_CV_CCV_TTS without OOC words results in extra insertions as discussed before, leading to more than 100% recognition of IV words. Our approach of synthesis based augmentation thus improves the recognition of OOV and IV words, mainly OOV, leading to improved WER.

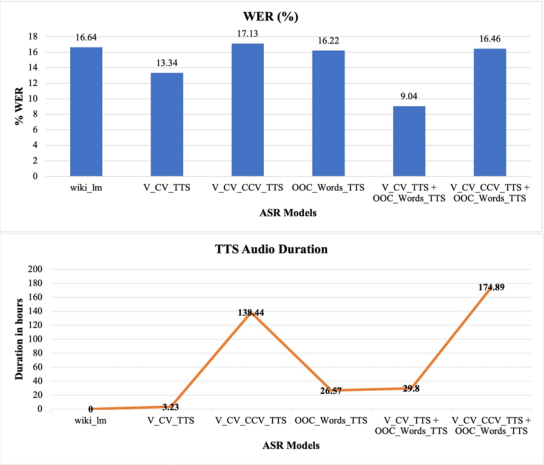


Fig 3. WER and corresponding TTS audio durations

Table 5. Percentage of OOV and IV words recognized

ASR Model	% OOV recognized	% IV recognized
wiki_aug_lm	62.28	94.59
V_CV_TTS	84.79	94.95
V_CV_CCV_TTS	84.44	101.2
OOC_Words_TTS	83.87	78.74
V_CV_TTS + OOC_Words_TTS	90.36	97.91
V_CV_CCV_TTS + OOC_Words_TTS	91	98.3

3.4 Comparison with other works

We compare our results with the works for Kannada language in [8,12], and the work for Tamil language in [11] as shown in Table 6. As depicted in Table 6, we obtain a WER of 9.04% for only 4 hours of training data using our approach which is comparable to the performance in [8] that uses 25 hours of training speech. The WER obtained by our method on 4 hours of continuous speech surpasses the results in [11,12].

Table 6. Comparison with existing works

Research	Language	Duration of training speech (hours)	WER (%)
[11]	Tamil	3.74	27.6
[8]	Kannada	25	4.05
[12]	Kannada	4	32.04
Our work	Kannada	4	9.04

4 Conclusion

Our empirical study regarding language model augmentation and lattice rescoring for low resource ASR depicts the following:

(i) Lattices generated by decoding with only the baseline language model may not contain all the probable words, lattice rescoring with a large language model (Wikipedia augmented) only adjusts the probabilities on the existing path because of which there is no significant improvement in OOV recovery as well as WER

(ii) Our approach of initial decoding with a baseline language model that is augmented to include unigram counts of OOV words (words that are present in the bigger text corpus but not in the baseline vocabulary) improves OOV recovery and hence enables inclusion of more words in the lattices, which when rescored with a large Wikipedia augmented language model results in a significant reduction of 16.68% in WER. This reduction is comparable to that of decoding with a larger language model, and our approach is memory efficient and requires only 2 GB memory while the memory required to decode with Wikipedia augmented language model is 18 GB.

The use of speech synthesis has been beneficial in improving ASR performance and is an area of research that is quite recent. The results obtained by our approach of syllable-based synthesis for data augmentation demonstrate the following – (i) inclusion of synthesized basic syllable sounds of a language improves the recognition capability of the ASR and reduces the WER by 10.91% (ii) augmenting with consonant conjuncts is more effective when combined with OOV words due to the presence of context for the consonants at the beginning of the conjuncts and (iii) the inclusion of synthesized syllables along with synthesized OOV words gives better improvement than the inclusion of synthesized OOV words alone. Combination of synthesized basic syllables and words containing OOC syllables (Model V_CV_TTS + OOC_Words_TTS) results in the best improvement of 15.21% in WER. Our synthesis based approach improves the recognition of both OOV (by 27%) and IV words (by 3%). We obtain the best accuracy of 90.96% (9.04% WER) overall using both language model augmentation and speech synthesis based augmentation.

In the future, we plan to apply our data augmentation approach to speech corpora in other low resource Indic languages (available at http://www.openslr.org/resources.php). In this study, we employed cross-lingual Telugu voices for speech synthesis in Kannada language. We wish to study the effect of borrowing sounds from other Indic languages for speech synthesis.

Acknowledgement

We thank Dr K.V. Subramanian, Head, Center for Cloud Computing and Big Data, PES University, Bangalore for all the support.

References

1) Pratap V, Sriram A, Tomasello P, Hannun A, Liptchinsky V, Synnaeve G, et al. Massively Multilingual ASR: 50 Languages, 1 Model, 1 Billion Parameters. *Electrical Engineering and Systems Science*. 2020. Available from: https://doi.org/10.48550/arXiv.2007.03001.
2) Office of the Registrar General I (ORGI), Commissioner C. Census of India. 2011. Available from: https://censusindia.gov.in/census.website/data/census-tables.
3) Srivastava BML, Sitaram S, Mehta RK, Mohan KD, Matani P, Satpal S, et al. Interspeech 2018 Low Resource Automatic Speech Recognition Challenge for Indian Languages. *6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*. 2018;p. 11–15. Available from: https://www.microsoft.com/en-us/research/publication/interspeech-2018-low-resource-automatic-speech-recognition-challenge-for-indian-languages/.
4) Sethi N, Dev A. Survey on Automatic Speech Recognition Systems for Indic Languages. *Artificial Intelligence and Speech Technology*. 2022;p. 85–98. Available from: https://doi.org/10.1007/978-3-030-95711-7_8.
5) Shinde AK, R AH, Karanth DN, K G, S VT. Development of Automatic Kannada Speech Recognition System . 2019. Available from: http://ijariie.com/AdminUploadPdf/Development_of_Automatic_Kannada_Speech_Recognition_System_ijariie10201.pdf.
6) Yadava GT, Jayanna HS. Automatic Isolated Kannada Speech Recognition System under Degraded Conditions. *International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques*. 2019;p. 146–150. Available from: https://doi.org/10.1109/iceeccot46775.2019.9114658.
7) Student P, Professor A, , , and. Kannada Speech Segmentation And Recognition For Speech To Text Conversion. *International Journal of Creative Research*. 2019;8(6):2320–2882. Available from: https://doi.org/10.1109/ICEECCOT46775.2019.9114658.
8) Kumar P, Jayanna HS. Development of Speaker-Independent Automatic Speech Recognition System for Kannada Language. *Indian Journal of Science and Technology*. 2022;15(8):333–342. Available from: https://doi.org/10.17485/IJST/v15i8.2322.
9) Kumar P, Yadava T, Jayanna HS. Continuous Kannada Speech Recognition System Under Degraded Condition. 2019. Available from: https://doi.org/10.1007/s00034-019-01189-9.
10) Yadava GT, Nagaraja BG, Jayanna HS. Enhancements in Continuous Kannada ASR System by Background Noise Elimination. 2022. Available from: https://doi.org/10.1007/s00034-022-01973-0.
11) Chellapriyadharshini M, Toffy A, M SRK, Ramasubramanian V. Semi-supervised and Active-learning Scenarios: Efficient Acoustic Model Refinement for a Low Resource Indian Language. *Interspeech* . 2018;p. 1041–1046. Available from: https://doi.org/10.48550/arXiv.1810.06635.
12) Murthy S, Sitaram D, Sitaram S. Effect of TTS Generated Audio on OOV Detection and Word Error Rate in ASR for Low-resource Languages. *Interspeech* . 2018;p. 1026–1056. Available from: https://www.microsoft.com/en-us/research/uploads/prod/2018/06/Effect-of-TTS-Generated-Audio-on-OOV-Detection-and-WER-in-ASR_revised_v1.pdf.
13) Zhang X, Povey D, Khudanpur S. OOV Recovery with Efficient 2nd Pass Decoding and Open-vocabulary Word-level RNNLM Rescoring for Hybrid ASR. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020;p. 6334–6342. Available from: https://doi.org/10.1109/ICASSP40776.2020.9053872.
14) Chen Z, Rosenberg A, Zhang Y, Wang G, Ramabhadran B, Moreno PJ. Improving Speech Recognition Using GAN-Based Speech Synthesis and Contrastive Unspoken Text Selection. *Interspeech* . 2020;p. 556–560. Available from: https://doi.org/10.21437/Interspeech.2020-1475.
15) Pusateri E, Van Gysel C, Botros R, Badaskar S, Hannemann M, Oualil Y, et al. Connecting and Comparing Language Model Interpolation Techniques. *Interspeech* . 2019. Available from: https://doi.org/10.48550/arXiv.1908.09738.
16) Manjunath KE, Jayagopi B, Rao D, Ramasubramanian KS. Articulatory-feature-based methods for performance improvement of Multilingual Phone Recognition Systems using Indian languages. 2020. Available from: https://doi.org/10.1007/s12046-020-01428-9.