Interspeech 2022
18-22 September 2022, Incheon, Korea

Using cross-model learnings for the Gram Vaani ASR Challenge 2022

Tanvina Patel and Odette Scharenborg

Multimedia Computing Group, Delft University of Technology, Delft, The Netherlands
t.b.patel@tudelft.nl, o.e.scharenborg@tudelft.nl

Abstract

In the diverse and multilingual land of India, Hindi is spoken as a first language by a majority of its population. Efforts are made to obtain data in terms of audio, transcriptions, dictionary, etc. to develop speech-technology applications in Hindi. Similarly, the Gram-Vaani ASR Challenge 2022 provides spontaneous telephone speech, with natural back-ground and regional variations in Hindi. The challenge provides: 100 hours of labeled train-set, 5 hours of labeled dev-set and 1000 hours of unlabeled data-set. For the 'Closed Challenge', we trained an End-to-End (E2E) Conformer model using speed perturbations, SpecAugment techniques and use VTLN to handle any unknown speaker groups in the blind evaluation set. On the dev-set, we achieved a 30.3% WER compared to the 34.8% WER by the Challenge E2E baseline. For the 'Self Supervised Closed Challenge', a semi-supervised learning approach is used. We generate pseudo-transcripts for the unlabeled data using a hybrid TDNN-3gram LM model and trained an E2E model. This is then used as a seed for retraining the E2E model with high confidence data. Cross-model learning and refining of the E2E model gave 25.3% WER on the dev-set compared to ∼33-35% WER by the Challenge baseline that use wav2vec models.
Index Terms: Gram-Vaani Challenge, end-to-end ASR, hybrid ASR, cross-architecture learning, semi-supervised learning

1. Introduction

India is a home to over 22 officially recognized languages and many other languages and dialects [1]. Indian languages are categorised into Indo-Aryan, Dravidian languages, Austroasiatic, Sino-Tibetan, Tai-Kadai, etc. language groups, and a few other minor language families [1, 2]. With the growing use of the Internet and with the spread of digitization in India, speech technology applications, Voice User Interface (VUI) devices in Indian languages will play a crucial role in the agriculture, health care, government sectors [3]. Recent advancements in speech technology have shown that Automatic Speech Recognition (ASR) systems can work on par with humans for read speech and with an in-domain and similar test-set [4, 5]. However, building such ASR systems and speech technology solutions requires large amounts of training data which are not sufficiently available for Indian languages.

Efforts have been made to collect data in Indian languages for speech recognition applications [6, 7, 8]. In the INTERSPEECH 2018 MSR low-resource challenge, read speech data was released in Gujarati, Tamil and Telugu language to build ASR systems [9, 10]. Several approaches were proposed that explore the similarities across the Indian languages by using multilingual training while also exploiting the unique properties of the target languages [11, 12, 13]. Similar initiatives are made to open source Hindi data as part of ASR Challenges and releasing data with increasing complexity in each challenge [14]. Inline to this, the Gram Vaani Hindi ASR Challenge 2022 is organized [15]. 'Gram Vaani' roughly translates to 'Rural Voice'. Considering the lower literacy level of the people belonging to the rural areas, developing speech applications can be of great benefit to them. As part of the Gram Vaani challenge, spontaneous telephonic speech with natural background and regional speech variations was released. The nature of the data make it a unique corpus for speech recognition in real-life scenarios.

Speech recognition approaches have been dominated by the hybrid Acoustic Model and Language model (AM-LM) approach and End-to-End (E2E) architectures. Both approaches have their pros and cons related to factors such as out of domain performance, OOV words, performance on long/short speech files, robustness, and real-time factor. As far as performance is considered, given sizeable data, E2E models have come to outperform hybrid models especially for spontaneous, telephone and noisy speech [16]. Hybrid ASR models are known to perform well in case of in-domain text that has a structure and where the LM plays a role. Hence, for low-resource settings it is worth exploring techniques that can combine benefits from different approaches. That is, ensemble approaches to combine the predictions of different approaches to improve robustness [17, 18]. Given a significant amount of unlabeled data, Semi-supervised Learning (SSL) is a known training approach [19, 20]. In a previous work with unlabeled noisy, telephone data, SSL has shown to improve performance in an E2E framework [21]. In this work we use SSL in a cross-model learning approach so to use the benefits of the hybrid model as well.

In the Gram Vaani challenge, we are provided with spontaneous labeled speech data for training and development, an unlabeled data-set and an unknown blind set for evaluation. Since E2E models generalize well to spontaneous and out-domain data, all our submitted models are E2E based. For the 'Closed' category that is supposed to be trained on only the given labeled data, we submit an E2E model with well-known techniques like speed perturbations and SpecAugment and add Vocal Tract length Normalization (VTLN) [22] features to take care of unseen speaker characters while evaluation on the blind-set. The E2E techniques are also data hungry and need more training data. Hence as a part of the 'Self Supervised' category that uses both the labeled and unlabeled data, we combine the strengths of both hybrid and E2E models in a semi-supervised approach. The hybrid ASR is used to get pseudo-transcripts of the unlabeled data and these are used to train an E2E model. This model is then used as a seed and, we incorporate SSL in the E2E framework. That is, utterances from the unlabeled corpus are decoded again with this seed and the hypotheses with higher confidence scores are further used to refine the seed model [23, 24]. This proposed approach as shown in Figure 1 has proved to be effective in reducing the Word Error Rate (WER) significantly as compared to that provided by the Challenge baselines.

2. Gram Vaani ASR Challenge 2022

This section discusses Gram Vaani ASR database details, various submission tracks and the Challenge baselines models [15].

10.21437/Interspeech.2022-10639

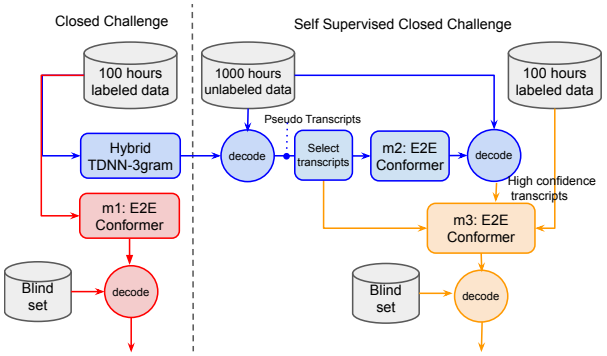

Figure 1: *Proposed Approach, Red: E2E system for Closed category, blue: pseudo-transcripts from hybrid ASR for better E2E seed, orange: retrain E2E model for Self Supervised category.*

2.1. Details of the Data-set

The details of the released data-set are shown in Table 1. The original data had .mp3 files with a mix of sampling rates ranging from 8 kHz to 48 kHz for the ∼1111 hours of data. The recordings were collected through the Mobile Vaani platform having users from all across India and, hence, it includes regional/dialectal variations of Hindi [15]. The speech is spontaneous with a natural background, making the data suitable for training real world applications. The transcriptions were done by crowd workers recruited via the Uliza platform and hence, may have varying degrees of accuracy.

Table 1: *Details of the Gram Vaani Challenge data-set*

Labeled (segment (Hours))		Unlabeled (segment (Hours))	
Train	**Dev**	**Other**	**Eval**
37152(100h)	1885(5h)	69591*(1000h)	1032(3h)

*1000 hours data consists of complete utterances and is not segmented

2.2. Challenge Submissions

A blind test set is released and participants are expected to submit the ASR hypotheses in any or all of these tracks on this set:

- *Closed Challenge*: Use only the Gram Vaani 100 hours train-set and 5 hours dev-set for training models. No pre-trained models or external data allowed.
- *Self Supervised Closed Challenge*: Use only the Gram Vaani 100 hours train-set, 5 hours dev-set and 1000 hours unlabeled data-set for training. No pre-trained models or external data allowed.
- *Open Challenge*: Use the Gram Vaani data-set and/or any other additional data-set or model.

2.3. Baseline Models

The organizers provided several baseline systems trained on the 100 hours labeled training data. The best-performing baselines, shown in Table 2, is a hybrid TDNN-3gram LM based ASR trained using Kaldi toolkit [25] and the E2E conformer architecture based ASR trained with ESPnet toolkit [26]. In addition, several wav2vec [27] models using other pre-trained models or using 1000 hours of data for pre-training and fine-tuning on 100 hours labeled data-set were also provided. These results were not as good as those of the hybrid and E2E ASRs despite adding more data for learning. One of the reasons could be that majority of the available data is sampled at 8 kHz, however, the baseline models are trained at 16kHz (i.e., possibly to match with the pre-trained models being available at 16kHz). The models submitted by the participants as part of the challenge are expected to beat these baseline systems in terms of error rates.

Table 2: *Performance of the Challenge baselines on dev-set*

Framework	Training	AM — LM	Dictionary	%WER
Kaldi	100 hrs	TDNN- 3gram LM	word	30.12
ESPnet	100 hrs	Conformer-No LM	BPE:1000	34.80

Pre-training-Wav2Vec 2.0 Base	Fine-tuning data	%WER
1000 hours Gramvaani unlabeled	100 hours labeled	35.97
pre-trained model: AI4Bharat	100 hours labeled	33.30
pre-trained model: Open-Speech-EkStep	100 hours labeled	34.32

3. The TU Delft Submission: Methodology

For the challenge we submit in the *Closed* and *Self Supervised Closed Challenge*. The details of the system are discussed next.

3.1. Closed Challenge Submission

3.1.1. Analysis on the Labeled Data

From the data analysis it is observed that more than 60% of the data is recorded at 8 kHz, this is likely due to the fact that the data consists of telephone speech. The 100 hours train-set contained utterances ranging in duration from 0.8 sec to 140 sec while the dev-set files ranged from 0.8 sec to 30 sec. Due to crowd sourcing, the transcripts might have errors, and indeed a few errors were observed in the training data. To analyse the transcript accuracy we decoded the training data with an in-domain AM and a biased LM using built-in kaldi functionalities [28, 29]. The AM-LM hybrid ASR system is discussed in Sec. 3.1.2. The edit distance between the reference and the decoded output by the biased LM can possibly be an estimate of the accuracy of the ground truth or reference. Figure 2 shows that most of the data, i.e., ∼95% of the data have an edit score of range 0-5 while the remaining ∼5% of the data have edit score in the range of 6-40, indicating that the latter transcriptions might be erroneous. An example of this is shown below where the decoded hypothesis by the biased LM is better than the reference transcription. Listening to this speech file, we found that the speech was hard to understand in some regions and the audio was clipped, probably being the cause for the poor transcription.

```
audio_5bf1f1da38dc02_640106410_23290300_1542582746
REF:खबर हाँ आयुष्मान भारत की मरीजों के लिए राहत भरी खबर हाँ
HYP:खबर हैं भारती आयुष्मान भारत के लिए राहत भरी खबर हैं जल्द ऐसी करने जा रहा है
EDITSCORE = 11
```

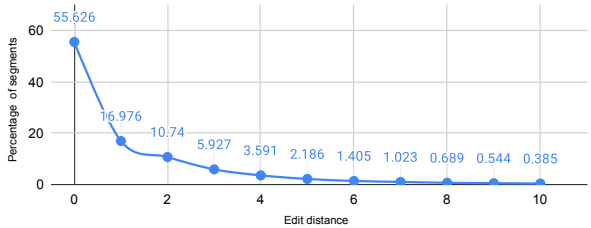

Figure 2: *Edit distance score between the manual transcripts of the 100 hours labeled data and decoded text using a biased LM*

3.1.2. Experiments on the Labeled Data

Based on the hypothesis that E2E models are suitable for spontaneous speech, we created several E2E models with different training strategies and compare them with the Challenge hybrid ASR system, which is the best baseline system in the Closed

category. The E2E ASR system consisted of the conformer architecture [30], and was trained using the ESPnet Framework [26]. State-of-the-art speed perturbations [31] and SpecAugment [32] techniques were applied. Since the blind test set may have speaker groups not seen in the training data, we also incorporate VTLN features [22] during E2E model training to help normalize speaker-specific characteristics. Additionally, we re-implemented the Challenge hybrid ASR baseline, i.e., TDNN with a 3-gram LM model trained using Kaldi [25]. Here, for the LM training, we randomly picked data from the train-set for validation, so that the LM is not biased to the dev-set.

Hindi has an almost one (character)-to-one (sound) correspondence, we therefore investigate the performance of the E2E model with different dictionaries. We train three systems: a character-based system with around 72 unique characters, a byte pair models with 1000 tokens, and other with 5000 tokens for training [33]. Any English text in the transcripts is removed and any text marked as 'incomplete text' is replaced by [UNK]. As the majority of the data is at 8 kHz, model training is carried out at this sampling rate, despite the baseline systems being trained at 16kHz. Details of training parameters are mentioned[1].

Table 3: *Results on Gram Vaani 5hrs dev-set using labeled data*

Training Details on 100hrs data		**%WER**			
AM — LM	**Dictionary**	**P0**	**P1**	**P2**	**P3**
TDNN-3gram LM	word	30.7	-	-	-
Conformer-No LM	character	36.3	34.5	31.4	31.1
Conformer-No LM	BPE:1000	34.0	32.3	30.5	**30.3**
Conformer-No LM	BPE:5000	33.1	33.9	30.4	**30.3**

P0: Labeled training data (No Augmentation)
P1: P0+SpecAugment
P2: P0+SpecAugment+Speed Perturbation
P3: P0+SpecAugment+Speed Perturbation+VTLN

Experimental results with our hybrid and E2E approaches evaluated on the 5 hours dev-set are shown in Table 3. Our hybrid system with the same architecture as the challenge baseline achieved a WER of 30.7% which is similar to the challenge baseline. Regarding the E2E models, the results show that using byte-pairs outperformed a character-based dictionary, with a better performance for the conformer model with 5k byte pairs (P0). However, using SpecAugment and speed perturbations for data augmentation removed the performance gap between the two byte pair models. Next, VTLN also showed a minor improvement in the performance. The best performing E2E model had 30.3% WER which is better than the 34.8% WER reported by the Challenge baseline using the same E2E architecture as shown in Table 2. The E2E model is also sightly better than our hybrid model. Hence, we use the E2E conformer model with 1k tokens and P3 parameters for the Closed Challenge and name this model $m1$ for future reference (shaded in Table 3).

As shown in Figure 2, about 5% of the training data had edit score > 5 when compared to the decoded transcripts with a biased LM. We ran a few data selection experiments to investigate if choosing the transcription that is longest improves the performance (i.e., it may either be the reference transcription or the decoded from the biased LM). However, the results showed that the WER for the best E2E model ($m1$) increased from 30.3% to 30.7% and that of the hybrid ASR model from 30.7% to 31.02%. Other variants were also tried like using only 95% of the lower edit score data, using the transcripts only from the biased LM ASR, however, in any case, the performance did not improve. Hence, in this work, we retain all of the training data despite the errors in the transcriptions. A more methodological way to select the accurate transcriptions for the poorly labeled data could be interesting to explore in future.

[1] $sample\text{-}freq$=8000, $num\text{-}mel\text{-}bins$=40, $bpemode$=$unigram$, $nbpe$=1000, $epochs$=30, $patience$=5, $n\_average$=5

3.2. Self Supervised Closed Challenge Submission

3.2.1. Analysis and Processing on the Unlabeled Data

The unlabeled data provided for the challenge also consists of .mp3 files with a mix of sampling rates ranging from 8 kHz to 48 kHz, with around 60% of the data at 8kHz. Unlike the labeled training data, the 1000 hours unlabeled data were not segmented, with audio files in the range of 10 sec to 3 minutes. Longer duration of audio files than those in the training are not well decoded by E2E models, and at times may yield out of memory issues. Hence, to use the data with E2E models, it is essential to split the audio. There are two ways to achieve this:

- Split the audio files into hard-splits of 20-30 sec, however, this may result in a split in-between words and may not always give a proper semantic split.
- Use a Voice Activity Detector (VAD) to segment the data based on the silence regions.

Hence, to split the unlabeled data, we use an energy-based VAD and generate the segments [25]. This generates ∼300k splits with a duration of 0.25 sec to 30 sec suitable for E2E models.

3.2.2. Cross-model learning

To maximally use an unlabeled data corpus, semi-supervised learning (SSL) is an efficient training approach [20], [23]. Here our goal in cross-model learning is two fold:

- Train an E2E model by using pseudo-transcripts of the unlabeled split data generated with a hybrid model
- Use the above trained model as a seed for refining or retraining the E2E model.

In the process of generating pseudo-transcripts we also ensure that transcriptions with a reasonable accuracy are used. Hence, we filter erroneous data by the same approach as in Section 3.1.1, i.e., again decode with an in-domain AM, and a biased LM. The hybrid model used is the TDNN-3gram LM model trained on labeled data. The first decoding gives pseudo reference transcriptions. The second decoding with a more weighted LM, is used to get the edit score. We assume that an 0 edit score is an indication that the transcripts are more or less correct and grammatically reliable. Around ∼200k segments corresponding to ∼870/1000 hours of unlabeled data had 0 edits.

The 0 edit pseudo transcript data is added to the original 100 hours of labeled data using the same training parameters of SpecAugment and VTLN as for the Closed Challenge. We did not apply speed perturbation to the 0 edit pseudo transcriptions of the unlabeled data to avoid multiplication of errors if any in the pseudo version. The results of this training are available in Table 4. It is observed that the approach gives a substantially lower WER of 25.3% on the dev-set. We name this model $m2$ (shaded row 1 in Table 4). To see the effect of adding more data, we also added the pseudo-transcripts with an edit score of 1, which yielded an almost identical WER of 25.4% WER on the dev-set. The small increase in WER might be due to the errors in the transcriptions. Overall, the WER is substantially less than the 33-35% WER reported by the challenge baseline models using the wav2vec approaches with pre-training and/or fine-tuning in the Self Supervised Category.

3.2.3. E2E Model Retraining

Once we have a robust seed model, we can use it to transcribe the unlabeled data again. Decoded utterances from the unlabeled corpus are then grouped into bins based on the confidence scores. Let the bins be denoted as $B_n, n = 1, ..., N$ defined by the confidence levels $(0.9, 1), (0.9, 0.8), ..., (0, 0.1), for N = 10$. The data with higher confidence is then used to refine or re-train the seed model until the WER improves. That is, the seed model is trained with data from each bin $B_n, n = 1, ..., N$ until the error drops after which the best model AM_n is chosen.

Implementing the entire iterative semi-supervised framework requires several rounds of training and hence, involves time and large computational resources. We therefore decide to re-train the seed $m2$ with bins B_1, B_2 and B_3 at once, i.e., corresponding to 1-0.7 confidence scores. From the results in Table 4, it is observed that the WER of the retrained model is slightly better than $m2$. The only small improvement even after adding significant amounts of data might be due to the model already having learned the data and is now being saturated. However, since the model has seen more data it should be more robust. We name this model as $m3$, (shaded row 3 in Table 4).

Table 4: *Results on Gram Vaani 5hrs dev-set using the labeled data, pseudo-labels (from the hybrid model) and the high confidence segments (from the E2E model) from the unlabeled data*

	Training Data Details	Hours	%WER
Conformer-No LM	Challenge data+ 0 edit score transcripts from hybrid model	100+ 872	25.3
	Challenge data+ 0&1 edit score transcripts from hybrid model	100+ 919	25.4
	Challenge Data+ 0 edit score transcripts from hybrid model+ B1-B3 bin data from E2E model	100+ 872+ 805	**25.0**

Finally, we estimate the confidence of each of the best models while decoding the unlabeled data, which is shown in Figure 3. For model $m1$ that was trained on only 100 hours of labeled data, there are only ∼2000 segments in bin B_1, i.e., with a confidence score of 1-0.9. The majority of the decoded utterances had a confidence score between 0.8-0.6. For $m2$, confidence for the decoded utterances was much higher, and a significant amount of data is decoded with high confidence in bin B_1, and the count continuously decreases for higher bins. The dotted line for $m3$ shows that there are slightly more utterances in the higher bin and hence, it indeed decodes with more confidence.

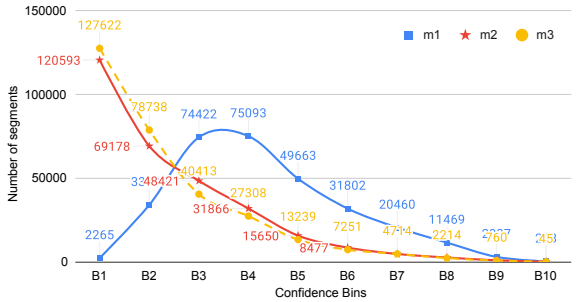

Figure 3: *Number of segments in each bin for $m1$, $m2$ and $m3$*

3.3. Submissions for the Blind-set

As per the Challenge guidelines, 3 submissions are allowed in each track. Here, we present our first submission in each track, while leaving remaining submissions for other experiments.

Closed Challenge: For this category, we submit model $m1$, i.e., the E2E model trained on labeled data with SpecAugment, speed perturbation and VTLN.

Self Supervised Closed Challenge: For this category, we submit model $m3$, i.e., the E2E model trained using labeled data, pseudo-transcripts with 0 edit score from the hybrid model, and higher confidence data from the E2E model

Table 5: *Results on the Gram Vaani blind-set*

Category:Model	AM — LM	%WER	%CER
Closed:m1	Conformer-No LM	30.43	16.74
Self-Supervised:m3	Conformer-No LM	**26.34**	**13.42**

Table 5 shows the leaderboard results on the blind-set in terms of Word Error Rate (WER) and Character Error Rate (CER) [15]. The baseline model for the Closed Challenge gives 29.7% WER (15.1% CER). At the time of submission of this paper, our $m1$ model was at second position, with a 30.43% WER (16.74% CER) as compared to the 29.34% WER (15.69% CER) from the first ranked SRI-B submission. In the Self Supervised category, our $m3$ with a 26.34% WER (13.42% CER) outperformed the Challenge baseline having a 31.83% WER (17.32% CER) in this category. Therefore, our proposed idea of cross-model and semi-supervised learning was able to improve the ASR performance. In the final results our $m3$ model stands at third position in terms of CER amongst all the submissions.

On listening to several files in the blind-set, we find that some of the files were difficult to understand and transcribe without the context. We also came across a child speech (segment 01-04927-02) in the blind-set which allows us to investigate the importance of VTLN while training. For this segment, the original transcript, the decoded transcript with and without VTLN using $m1$ model parameters is as follows:

```
m1 (novtln) : इस तीरंगे झंडे की धूमकान
m1 (vtln)   : इस तिरंगे झंडे की गुमगान धन्यवाद
Original    : इस तिरंगे झंडे की गुणगान अन्य
```

For the without-VTLN model, the highlighted text goes wrong while the VTLN model transcript is close to original text. Thus, adding VTLN might handle speaker characteristics better.

4. Conclusions and Future Directions

In this work, we used semi-supervised learning technique to deal with unlabeled data of the Gram Vaani Hindi Challenge. We apply cross-model training approach for E2E models to incorporate the benefits of the hybrid model. The E2E models trained on labeled data performed equivalent to hybrid models; however, when E2E models were trained with additional, selected unlabeled data with cross-model retrieved pseudo-transcripts, E2E models gave a significant improvement.

The E2E retraining approach did not show significant improvement over learning from pseudo-transcripts from the hybrid model and hence, we would like to explore other complimentary information, say possibly using pseudo-transcripts from transformers architecture for semi-supervised learning. We intend to develop parallel systems with the hybrid models for comparisons. Moreover, in addition to using confidence measures, we would like to explore a way to choose the best transcript from different architectures. This would reduce annotation time by many folds and if active learning is incorporated, the low confidence data can be annotated and used for training.

5. References

[1] Wikipedia. (2022) Languages of India. [Online]. Available: https://en.wikipedia.org/wiki/Languages_of_India

[2] N. W. Encyclopedia. (2018) Languages of India. [Online]. Available: http://www.newworldencyclopedia.org/entry/Languages_of_India

[3] A. Mohan, R. Rose, S. H. Ghalehjegh, and S. Umesh, "Acoustic modelling for speech recognition in Indian languages in an agricultural commodities task domain," *Speech Communication*, vol. 56, pp. 167–180, Jan 2014.

[4] D. Amodei and et al., "Deep Speech 2: End-to-end speech recognition in English and Mandarin," in *Proc. of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ser. ICML, 2016, pp. 173–182.

[5] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[6] M. Kumar, N. Rajput, and A. Verma, "A large-vocabulary continuous speech recognition system for Hindi," *IBM Journal of Research and Development*, vol. 48, no. 5-6, pp. 703–715, 2004.

[7] G. A. Numanchipalli, R. Chitturi, S. Joshi, R. Kumar, S. P. Singh, R. Sitaram, and S. P. Kishore, "Development of Indian language speech databases for large vocabulary speech recognition systems," in *SPECOM*, Patras, Greece, 2005, pp. 1–5.

[8] T. Patel, K. DN, N. Fathima, N. Shah, M. C, D. Kumar, and A. Iyengar, "Development of large vocabulary speech recognition system with keyword search for Manipuri," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 1031–1035.

[9] B. M. L. Srivastava, S. Sitaram, R. K. Mehta, K. D. Mohan, P. Matani, S. Satpal, K. Bali, R. Srikanth, and N. Nayak, "Interspeech 2018 low resource automatic speech recognition challenge for Indian languages." in *Spoken Language Technologies for Under-Resourced Languages (SLTU)*, August 2018, pp. 11–14.

[10] INTERSPEECH. (2018) Special Session: Low resource speech recognition challenge for Indian languages. [Online]. Available: https://www.microsoft.com/en-us/research/event/interspeech-2018-special-session-low-resource-speech-\\recognition-challenge-indian-languages/

[11] N. Fathima, T. Patel, M. C, and A. Iyengar, "TDNN-based multilingual speech recognition system for low resource Indian languages," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 3197–3201.

[12] J. Billa, "ISI ASR system for the low resource speech recognition challenge for Indian languages," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 3207–3211.

[13] B. Pulugundla, M. K. Baskar, S. Kesiraju, E. Egorova, M. Karafiát, L. Burget, and J. Černocký, "BUT system for low resource Indian language ASR," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 3182–3186.

[14] IITM. (2020) Automatic Speech Recognition (ASR) Hindi Challenge. [Online]. Available: https://sites.google.com/view/asr-challenge/home

[15] IITM. (2022) GRAM VAANI ASR Challenge 2022. [Online]. Available: https://sites.google.com/view/gramvaaniasrchallenge

[16] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, S. Watanabe, T. Yoshimura, and W. Zhang, "A comparative study on transformer vs rnn in speech applications," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Sentosa, Singapore, 2019, pp. 449–456.

[17] C. Li, F. Keith, W. Hartmann, M. Snover, and O. Kimball, "Overcoming domain mismatch in low resource sequence-to-sequence ASR models using hybrid generated pseudotranscripts," *CoRR: arXiv*, vol. abs/2106.07716, 2021.

[18] L. Deng and J. Platt, "Ensemble deep learning for speech recognition," in *Proc. Interspeech*, Singapore, 2014, pp. 1915–1919.

[19] Y. Zhang, J. Qin, D. S. Park, W. Han, C.-C. Chiu, R. Pang, Q. V. Le, and Y. Wu, "Pushing the limits of semi-supervised learning for automatic speech recognition," *arXiv preprint arXiv:2010.10504*, 2020.

[20] S. Karita, S. Watanabe, T. Iwata, A. Ogawa, and M. Delcroix, "Semi-supervised end-to-end speech recognition," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 2–6.

[21] Tanvina Patel, "Semi-Supervised Learning for speech recognition in Indian languages," in *NVIDIA GPU Technology Conference (GTC)*, October 2020, p. [A21560].

[22] D. Y. Kim, S. Umesh, M. J. F. Gales, T. Hain, and P. C. Woodland, "Using VTLN for broadcast news transcription," in *Proc. Interspeech*, Jeju Island, 2004, pp. 1953–1956.

[23] M. Chellapriyadharshini, A. Toffy, S. R. K. M., and V. Ramasubramanian, "Semi-supervised and active-learning scenarios: Efficient acoustic model refinement for a low resource Indian language," in *Proc. Interspeech*. Hyderabad, India: ISCA, 2018, pp. 1041–1045.

[24] A. Madan, A. Khopkar, S. Nadig, K. M. Srinivasa Raghavan, D. Eledath, and V. Ramasubramanian, "Semi-supervised learning for acoustic model retraining: Handling speech data with noisy transcript," in *International Conference on Signal Processing and Communications (SPCOM)*, Bangalore, India, 2020, pp. 1–5.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE Signal Processing Society, 2011.

[26] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-End speech processing toolkit," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 2207–2211.

[27] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

[28] V. Manohar, D. Povey, and S. Khudanpur, "JHU Kaldi system for Arabic MGB-3 ASR challenge using diarization, audio-transcript alignment and transfer learning," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Stockholm, Sweden, 2017, pp. 346–352.

[29] Kaldi Recipe Used. [Online]. Available: https://github.com/kaldi-asr/kaldi/blob/master/egs/wsj/s5/steps/cleanup/clean_and_segment_data_nnet3.sh

[30] A. Gulati, J. Qin, C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech*, Shanghai, China, 2020, pp. 5036–5040.

[31] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Proc. Interspeech*, Dresden, Germany, 2015, pp. 3586–3589.

[32] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[33] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proc of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725.