Interspeech 2024
1-5 September 2024, Kos, Greece

A Unified Approach to Multilingual Automatic Speech Recognition with Improved Language Identification for Indic Languages

Nikhil Jakhar, Sudhanshu Srivastava, Arun Baby

Samsung Research, Bangalore

n.jakhar@samsung.com, sudhanshu.sv@samsung.com, arun.baby@samsung.com

Abstract

Multilingual Automatic Speech Recognition (ASR) presents several difficulties, especially when multiple languages are being spoken in the same audio. Traditional multilingual ASR systems often rely on low-resource Indic language data and language-specific models, which limits their scalability and efficiency. Creating individual models is difficult due to the lack of Indic language data, while the need for an accurate language identification (LID) model further affects the downstream task. Our method integrates LID and multilingual ASR in a unified framework, leveraging their symbiotic relationship to overcome limitations. This study presents an approach to multilingual ASR incorporating LID capabilities using Whisper as the baseline architecture. Experimental results on benchmark datasets demonstrate our method's effectiveness, which shows an absolute 19.1% improvement in Word Error Rate (WER) while enhancing LID performance by 6% in terms of Diarization Error Rate (DER).
Index Terms: multilingual ASR, language identification, dispalce challenge

1. Introduction

Automatic Speech Recognition (ASR) system is one of the most prominent works in the field of Natural Language Understanding (NLU). It finds immense application in human-machine interaction and recent monolingual ASR systems like Bixby, Alexa and Siri demonstrate its application in day-to-day life.

Most people welcome ASR systems in their mother tongue. Traditionally, most works in the field have focused on increasing the accuracy and correctness of systems. Earlier works employed Hidden Markov model (HMM) - Gaussian Mixture model (GMM) based architectures. While these models were characterized by speed, resilience in low-resource environments, and a small footprint, the emergence of Deep Neural Network (DNN) based models, and more recently, End-to-End (E2E) models like cascaded encoders [1] have caused a shift in the paradigm with increased performance and low word-error-rate (WER).

A complete survey of the advancements of monolingual ASR systems is provided by Wang et al. in [2]. Post HMM-DNN based methods, the authors point how CTC, which essentially is a loss function enables fuller use of DNNs in ASR models. However, CTC is unable to model interdependencies within the output sequence and it only maps input sequences to output sequences with shorter lengths. Recurrent-Neural-Networks-Transducers (RNN-T) solve the aforementioned shortcomings. But, RNN-T is not easy to train and makes unreasonable calculations. Finally, it is shown that attention-based E2E models achieve better results than CTC and RNN-T.

India's linguistic landscape is notably diverse, with the census recognizing 1576 languages and 1796 mother tongues, of which 22 are official. Furthermore, there are over 3500 newspapers and periodicals published in more than 30 languages, with approximately 70 languages taught in schools. The radio network broadcasts programs in 146 languages and dialects. Within a population exceeding 1 billion, 26% are bilingual and 7% are trilingual [3].

This linguistic diversity gives rise to code-switching (CS), the practice of using two or more languages within a single utterance. This inherent diversity makes Indic languages digitally low resource, necessitating models capable of handling CS. Various works explore the possibility of a unified phone set across Indian languages to address this issue [4, 5]. Also, code-mixing has been experimented with in text-to-speech along with the help of a merged phone set [6]. Joint training of ASR and LID emerges as a cost-effective and efficient alternative, particularly benefiting languages with low amounts of data compared to others by pooling.

One of the first works in literature for joint recognition of speech and language without relying on language-specific modules such as dictionaries and language models pioneers the use of a hybrid attention/CTC module alongside a CNN+BLSTM encoder and RNN-LM decoder [7]. However, it falls short of effectively handling code-switching within utterances. To address this, the authors propose a solution in [8], where all characters from target languages are aggregated into a unified set. This augmented character set is then utilized for training, supplemented by the inclusion of language IDs in the token set, leading to the creation of a new corpus for multilingual speech recognition.

Attention-based models for recognizing Mandarin-English code-switched speech have been improved by incorporating three enhancements [9]. Firstly, the integration of multi-task learning for language identification was proposed. This improvement not only identifies suitable locations for applying LID loss but also predicts LID at the character level. Additionally, the second enhancement suggests utilizing wordpieces instead of graphemes to enhance modelling and minimize the gap in modelling units. Lastly, transfer learning was employed to leverage the existing monolingual training data.

Meanwhile, the RNN-T framework is adapted to jointly train ASR and LID models for Indian English and spoken Hindi [10]. The acoustic LID model is trained using cross-entropy loss, and its predictions are integrated into the joint ASR-LID network. A distinct approach was introduced by Lee et al. [11], where a unified pronunciation model across Korean and English is developed using phonetic information. The authors tackle the challenge of imbalanced training data due to a lack of CS data by performing language model domain adaptation on semanti-

10.21437/Interspeech.2024-2043

cally similar sentences of rare English words in Korean when using LM fusion.

The state-of-the-art (SOTA) LID-ASR model, termed Whisper, employs a Transformer-based architecture pre-trained on 680k hours of data, with the flexibility for fine-tuning in domain-specific tasks [12]. However, while highly effective for English, Whisper's performance diminishes significantly for Indian languages. We will see more on this in the following section.

For the Indic languages, work done by Punjabi et al. [10] discusses Indian English and Hindi. Furthermore, another approach by Chadha et al. [13] uses predictions from a multilingual model as LID to call individual monolingual models. Recent work in this domain focuses on leveraging linguistic knowledge to establish a common label set spanning various languages, primarily aiming at enhancing ASR performance [14].

Previous works for the multilingual Indic ASR systems include efforts to create new representation sets for labelling speech [14, 15, 16]. Similar to Jayakumar et al. [14], Kumar et al. [15] created a new labelling set and added graphemes to that set whereas Anoop et al. [16] leveraged the pronunciation similarity in South Indian languages and made a new set. A dataset consisting of 17,314 hours of raw audio data for pre-training across 40 languages was introduced by Javed et al. [17] and by conducting an ablation study on different architecture choices, they discerned that ASR accuracy for Indian languages fall behind those achieved for a well-resourced language like English.

To bridge this disparity and streamline model deployment, we propose a standalone model optimized for Indic languages, jointly trained for ASR and LID tasks without new speech labels. By fine-tuning the Whisper model with Indic languages such as Indian English, Hindi, Telugu, Bengali, and Kannada, tagged with their respective language identifiers, we aim to reduce the number of models required and overall storage footprint.

2. Proposed approach

Our proposed approach consists of two methods. We focus our effort on fine-tuning the Whisper model. The pre-trained Whisper model demonstrates a strong ability to generalise to different datasets and domains. However, its predictive capabilities can be improved further for certain languages and tasks through fine-tuning.

In the first approach (**Proposed-v1**), we fine-tune the whisper model using the Indic data collected from various resources. In the traditional fine-tuning approach, each language is fine-tuned separately. By combining ASR tokens with the appropriate language tokens in the loss function, we fine-tune all five languages in this approach. Here, all the fine-tuning parameters are kept as such in the given Huggingface transformers [1] repository. We have augmented the data in various ways for this. Utterances from different languages are joined together to augment the training data. While joining, the language token for the combined utterance is decided based on the majority token in the final utterance. This proposed system is depicted in Figure 1.

In the second approach (**Proposed-v2**), we experiment with weighted loss for the language identification tag and the ASR. In this way, we can control how well the language ID should be learned along with the ASR. The loss function for the first approach, Whisper fine-tuning is:

$$l_{tot} = \frac{l(LID_{pred}, LID_{true}) + \sum_{i=1}^{n} l(tok^{i}_{pred}, tok^{i}_{true})}{n+1} \tag{1}$$

Here, n represents the number of text tokens, l represents the cross-entropy loss function, LID_{pred} is the predicted LID, LID_{true} is the actual LID of the audio, tok^{i}_{pred} is the i^{th} predicted ASR token and tok^{i}_{true} is the actual ASR token. Additionally, l_{tot} represents the final loss for an audio-transcript pair.

This loss is modified to accommodate weightage and the modified loss function is:

$$l_{tot} = w_1 l(LID_{pred}, LID_{true}) + \frac{w_2 \sum_{i=1}^{n} l(tok^{i}_{pred}, tok^{i}_{true})}{n} \tag{2}$$

Where w_1 is the weight given to the LID token's loss and w_2 is the weight given to ASR tokens' loss

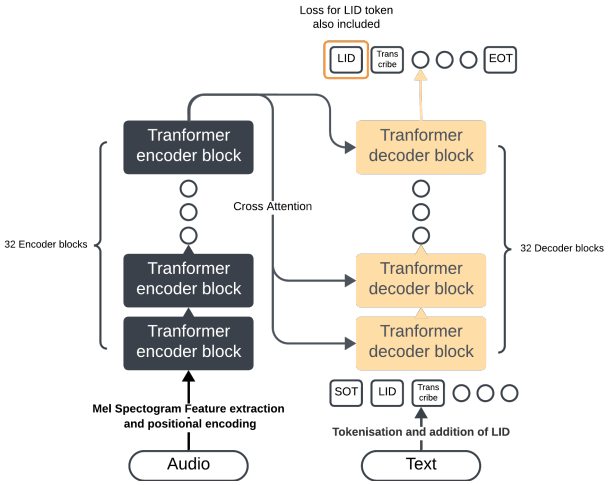

Figure 1: *Whisper large model architecture and flow*

3. Experimental setup

The dataset used, model parameters and the baseline setup are explained in this section. We have collected data from different sources and each of them is detailed. The Whisper model used for fine-tuning and the baselines used for all the experiments are described in detail.

3.1. Dataset used

We have used 3 close-field datasets for our Training; SPRINGINX, IndicTTS, and IndicVoices [18, 19, 20]. Languages used in our training are listed below:

- Indian English (En)
- Hindi (Hi)
- Bengali (Bn)
- Telugu (Te)
- Kannada (Kn)

[1] https://github.com/huggingface/transformers

SPRING-INX [2] [18]: This dataset consists of ASR data for Indic languages. The audios in the dataset have 16 kHz sampling frequency, collected from different noisy scenarios. The gender ratio is shown in Table 1.

Language	Male	Female
Bengali	936	797
Hindi	764	639
Kannada	200	281

Table 1: *Gender Balance in SPRING-INX Data. The numbers represent the speaker count.*

IndicTTS [3] [19]: It is an open-source database consisting of speech recorded at 48 kHz sampling frequency in a studio environment. It is to be noted that the style is read speech with grammatically correct sentences. These are primarily recorded for TTS and datasets of 13 Indian languages (monolingual + Indian English) are available.

Indic-Voices [4] [20]: This is 7348 hours of Indic data spoken by 16237 speakers out which 1639 hours are transcribed. We use Hindi, Bengali, Telugu and Kannada languages from the transcribed corpus. The amount of data (in hours) for each language used from this dataset is present in Table 2.

Dataset	En	Hi	Bn	Te	Kn
SPRING-INX	-	351	420	-	97
IndicTTS	302	-	-	17	-
IndicVoices	-	65	100	98	50

Table 2: *Dataset used for Track-2 and Track-3 (in hours)*

To reduce the disparity between the languages in training data for our proposed approach, we have employed noise augmentation for Telugu and Kannada so that all the languages represent an equal share in the training data.

For Track-2, the LID task, all evaluations are performed on the dev dataset provided by the DiSPLACE 2024 challenge [21]. This dataset contains 35 sessions of multi-speaker multi-lingual audio recordings at 16 kHz sampling frequency. Further, segmentation details are also given for each of these sessions. A total of 36000 segments were available for this LID task.

For the evaluations, we have used the DiSPLACE challenge 2024 [21] [5] dataset with non overlapping speakers compared to the training data. For the ASR task (Track-3) the dataset provided for evaluations comprises roughly 32 hours of conversational data, each conversation lasting between 30-60 minutes with 3 to 5 participants. Language wise details are given in Table 3.

Dataset	En	Hi	Bn	Te	Kn
dev	1.43	0.25	0.31	0.58	0.1

Table 3: *Track-3 Close-field Dev dataset details provided by DiSPLACE 2024 (in hours)*

3.2. Experimental details

For our experiments, we have chosen the Whisper model which is the state-of-the-art multilingual ASR model. The basic architecture of the Whisper model is shown in Figure 1. We chose, the Whisper-large-v3 model for the down-streaming tasks done in this study. It has 32 transformer encoder layers and 32 transformer decoder layers with a total of 1550M parameters [12]. It also boasts a vocabulary size of 51866 tokens which consists of tokens from several languages and special tokens like the language identification token. The input uses 128 Mel frequency bins extracted from the audio. The Whisper large-v3 model is trained on 1 million hours of weakly labelled audio and 4 million hours of pseudo-labeled audio collected using Whisper large-v2. The model was trained for 2.0 epochs over this mixture dataset.

This model was fine-tuned for single as well as multiple languages. During the fine-tuning on a dataset containing multiple languages, the language tag for each audio was also taken into account during loss calculation. An experiment that only computes loss based on the language tag was also done.

Apart from fine-tuning the model, we have tried using different combinations of datasets to validate the dataset's efficacy in improving the accuracy. In this, two models are trained, the first one with two datasets; SPRINF-INX data and IndicTTS data, and the second one with all 3 datasets mentioned in Table 2.

3.3. Baseline setup

To show the efficacy of our proposed method, we compared our approach to multiple baseline systems.

3.3.1. DiSPLACE challenge baseline

The Google Speech to Text [6] cloud services are used for our baseline system using the Close-field recordings of development data. The Speaker diarization is kept off and the respective language mode is used for the API call. Besides the above settings, we have used audio segments longer than or equal to 2 seconds while giving input to the API and computed the Word Error Rate (WER) for Close-field recording per language (by concatenating all the generated transcripts per language into one single file)

3.3.2. Whisper baseline

For this baseline, the Whisper model explained in Section 3.2 was used without fine-tuning.

3.3.3. Whisper fine-tuned for a single language

The model explained in Section 3.3.2 is fine-tuned with the respective language and Individual models are created for each of the 5 languages listed before.

4. Results and Discussion

Our proposed system is compared against the 3 baselines discussed in Section 3.3. The first results row represents the baseline system detailed in Section 3.3.1. The second results row represents the Whisper-based baseline system detailed in Section 3.3.2. In the 4^{th} and 5^{th} results rows, the evaluation results of models fine-tuned individually (Individual-FT) for each of the languages are shown. Finally, in the 6^{th} and 7^{th} the results of our proposed system are shown. For the Individual as well as the proposed systems, we have 2 variations based on the data used for the training. In version 1 (denoted as v1), we have used only SPRING-INX and IndicTTS datasets for training and for version 1.1 (denoted as v1.1) all 3 datasets detailed in Table 2

[2] https://asr.iitm.ac.in/dataset

[3] https://www.iitm.ac.in/donlab/indictts

[4] https://ai4bharat.iitm.ac.in/indicvoices/

[5] https://displace2024.github.io/

[6] https://cloud.google.com/speech-to-text

are used. In all the languages our proposed system outperforms the Baseline system with a weighted average of 19.1% WER improvement. However, In the case of low-resource languages like Telugu and Kannada, the individual fine-tuned model performs slightly better compared to our proposed system.

System	En	Hi	Bn	Te	Kn
Baseline	66.5	58.5	63.5	71.2	80.8
Whisper	49.6	62.8	116.8	114.9	179.5
Individual-FT-v1	32.3	39.7	56.0	97.7	67.6
Individual-FT-v1.1	-	36.6	55.1	**68.2**	**63.5**
Proposed-v1	**32.1**	41.5	**54.8**	107.9	91.5
Proposed-v1.1	32.5	**40.0**	55.9	**85.6**	**73.0**

Table 4: *Evaluation results (WER) on Close-field Dev set of Track-3 (lower is better)*

To understand the performance of the Individual-FT (Lang-FT) models, we evaluated three of the single-language fine-tuned models for languages other than the fine-tuned one. The results are shown in Table 5. Here, the baseline system considered is the Whisper model explained in Section 3.3.2. The proposed approach improves overall performance and outperforms the Whisper baseline in lesser resource languages. Owing to the huge size gap between the original train data for languages like Hindi and Bengali compared to Telugu and Kannada, the performance for these languages is poor. Adding the IndicVoices [20] dataset shows significant improvement across the languages for both Individual fine-tuned models and proposed systems.

Our proposed system performs better across all the languages in the multilingual scenarios as shown in Table 4. For the independent single-language fine-tuned models, the performance improvement for the fine-tuned language comes at a cost of the performance for the other languages as shown in Table 5. The performance of the Telugu and Kannada for multilingual models is slightly less than the Individually fine-tuned model, because of less data availability for those languages. Still, it outperforms the baseline Whisper model.

System	En	Hi	Bn	Te	Kn
Whisper	49.6	62.8	116.8	114.9	179.5
Bn-FT-v1	63.9	58.5	56.0	109.9	118.0
Bn-FT-v1.1	95.3	103.8	55.1	111.5	121.3
Te-FT-v1	129.4	169.1	173.7	97.7	154.8
Te-FT-v1.1	105.1	106.7	124.6	**68.2**	105.8
Kn-FT-v1	118.0	107.9	128.5	109.7	67.6
Kn-FT-v1.1	113.9	103.8	124.4	104.5	**63.5**
Proposed-v1	**32.1**	41.5	**54.8**	107.9	91.5
Proposed-v1.1	32.5	**40.0**	55.9	**85.6**	**73.0**

Table 5: *Evaluation results (WER) of single language fine-tuned models on Close-field Dev set of Track-3 (lower is better). Here it can be seen that all the languages except the one the model has been fine-tuned on degrade drastically.*

One interesting observation that we found is that while fine-tuning for either Telugu or Kannada, the performance of both languages is improved. This could be attributed to Telugu and Kannada belonging to the same language family, Dravidian and having phonetic similarities. Suggesting that fine-tuning for a language can improve the system for the languages in the same family.

To improve the language identification accuracy, we have fine-tuned our model based on the loss mentioned in Equation 1 and the results are shown in Table 6. Here the model is jointly fine-tuned for LID and ASR tasks. This gives a performance improvement from 24.04% to 18.79% DER. In the case of Proposed-v2 where the w_1 is 1.0, the fine-tuning is restricted to LID only. However, we have seen that LID only fine-tuning performs worse than the joint LID and ASR fine-tuning. Adding a LID token in the loss function improves the accuracy of LID as the model learns to distinguish between audios by learning the textual context

To understand the efficacy of the Proposed-v2 approach detailed in Section 2, we have experimented with varying weightage to loss function. This experiment was mainly to improve the LID performance of the Proposed system. The weightage given for the loss function explained in Section 2 is experimented with. The best result for our Proposed-v2 approach for LID is obtained when the LID loss is weighted at 0.8. However, the weightage of 1.0 for LID loss degraded the model overall. This says that the joint fine-tuning of the model is better compared to the LID-only fine-tuning.

System	w_1	w_2	DER
Whisper	-	-	24.04
Baseline	-	-	40.58
Proposed-v1	-	-	18.79
Proposed-v1.1	-	-	19.27
Proposed-v2	1.0	0.0	23.46
Proposed-v2	0.8	0.2	**18.03**
Proposed-v2	0.6	0.4	18.21
Proposed-v2	0.4	0.6	19.37
Proposed-v2	0.2	0.8	18.29

Table 6: *Evaluation results (DER) on Dev set of DISPLACE challenge Track-2 (lower is better).*

The IndicVoices [20] which has audio data for all 22 official languages of India, improved our system performance in all cases. This signifies that adding more relevant data can improve the performance of our proposed system. For the low-resource languages Kannada and Telugu, the improvements are significant after adding the IndicVoices dataset to both individual fine-tuned models as well as our proposed systems. For example, the WER for Telugu has decreased by 29.5% and 22.3% for the Individual and proposed system respectively. For Kannada, the improvement is 4.1% and 18.5% for the Individual and proposed system respectively.

5. Conclusion

In conclusion, this paper introduces a novel approach to multilingual Automatic Speech Recognition (ASR) that integrates language identification within a unified framework. By leveraging the symbiotic relationship between language identification and ASR, our method overcomes limitations associated with traditional multilingual ASR systems. The traditional approaches of singular model fine-tuning are inferior to our proposed approach. Also, the weighted loss function further improves our LID performance. Experimental results on benchmark datasets demonstrate a significant improvement in Word Error Rate (WER) by 19.1% and a 6% enhancement in language identification performance, validating the effectiveness of our approach. We have also shown that adding more relevant data, especially for the low-resource languages can bring much better performance to the proposed approach. This advancement holds promise for improving the scalability and efficiency of multilingual ASR systems, particularly in low-resource settings with Indic languages.

6. References

[1] A. Narayanan, T. N. Sainath, R. Pang, J. Yu, C.-C. Chiu, R. Prabhavalkar, E. Variani, and T. Strohman, "Cascaded Encoders for Unifying Streaming and Non-Streaming ASR," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5629–5633.

[2] D. Wang, X. Wang, and S. Lv, "An Overview of End-to-End Automatic Speech Recognition," *Symmetry*, vol. 11, no. 8, 2019. [Online]. Available: https://www.mdpi.com/2073-8994/11/8/1018

[3] G. Ministry of Education, "Languages in India." [Online]. Available: https://www.education.gov.in/sites/upload_files/mhrd/files/upload_document/languagebr.pdf

[4] A. Baby, N. NL, A. L. Thomas, and H. A. Murthy, "A unified parser for developing indian language text to speech synthesizers," in *TSD*, 2016.

[5] A. Baby, P. Jawale, S. Vinnaitherthan, S. Badam, N. Adiga, and S. Adavane, "Non-native English lexicon creation for bilingual speech synthesis," in *SSW 11*, 2021.

[6] A. L. Thomas, A. Prakash, A. Baby, and H. Murthy, "Code-switching in Indic Speech Synthesisers," in *Interspeech 2018*, 2018.

[7] S. Watanabe, T. Hori, and J. R. Hershey, "Language Independent End-to-End Architecture for Joint Language Identification and Speech Recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017, pp. 265–271.

[8] H. Seki, S. Watanabe, T. Hori, J. L. Roux, and J. R. Hershey, "An End-to-End Language-Tracking Speech Recognizer for Mixed-Language Speech," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4919–4923.

[9] C. Shan, C. Weng, G. Wang, D. Su, M. Luo, D. Yu, and L. Xie, "Investigating End-to-end Speech Recognition for Mandarin-english Code-switching," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6056–6060.

[10] S. Punjabi, H. Arsikere, Z. Raeesy, C. Chandak, N. Bhave, A. Bansal, M. Müller, S. Murillo, A. Rastrow, A. Stolcke, J. Droppo, S. Garimella, R. Maas, M. Hans, A. Mouchtaris, and S. Kunzmann, "Joint ASR and Language Identification Using RNN-T: An Efficient Approach to Dynamic Language Switching," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7218–7222.

[11] D. Lee, D. Kim, S. Yun, and S. Kim, "Phonetic Variation Modeling and a Language Model Adaptation for Korean English Code-Switching Speech Recognition," *Applied Sciences*, vol. 11, no. 6, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/6/2866

[12] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[13] H. S. Chadha, P. Shah, A. Dhuriya, N. Chhimwal, A. Gupta, and V. Raghavan, "Code Switched and code mixed speech recognition for Indic languages," Mar. 2022.

[14] K. Jayakumar, V. N. Sukhadia, A. Arunkumar, and S. Umesh, "The Tag-Team Approach: Leveraging CLS and Language Tagging for Enhancing Multilingual ASR," in *Proc. INTERSPEECH 2023*, 2023, pp. 4414–4418.

[15] M. G. Kumar, J. Kuriakose, A. Thyagachandran, A. K. A, A. Seth, L. V. D. Prasad, S. Jaiswal, A. Prakash, and H. A. Murthy, "Dual Script E2E Framework for Multilingual and Code-Switching ASR," in *Proc. Interspeech 2021*, 2021, pp. 2441–2445.

[16] C. S. Anoop and A. G. Ramakrishnan, "Exploring a Unified ASR for Multiple South Indian Languages Leveraging Multilingual Acoustic and Language Models," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 830–837.

[17] T. Javed, S. Doddapaneni, A. Raman, K. S. Bhogale, G. Ramesh, A. Kunchukuttan, P. Kumar, and M. M. Khapra, "Towards Building ASR Systems for the Next Billion Users," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, pp. 10 813–10 821, Jun. 2022. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/21327

[18] N. R, M. S, J. F, A. Gangwar, M. N. J, S. Umesh, R. Sarab, A. K. Dubey, G. Divakaran, S. V. K, and S. V. Gangashetty, "SPRING-INX: A multilingual Indian language speech corpus by SPRING Lab, IIT madras," Oct. 2023.

[19] A. Baby, A. L. Thomas, N. N. L, and H. A. Murthy, "Resources for Indian Languages," in *Community-based Building of Language Resources (International Conference on Text, Speech and Dialogue)*, 2016, pp. 37–43.

[20] T. Javed, J. A. Nawale, E. I. George, S. Joshi, K. S. Bhogale, D. Mehendale, I. V. Sethi, A. Ananthanarayanan, H. Faquih, P. Palit, S. Ravishankar, S. Sukumaran, T. Panchagnula, S. Murali, K. S. Gandhi, A. R, M. K. M, C. V. Vaijayanthi, K. S. R. Karunganni, P. Kumar, and M. M. Khapra, "IndicVoices: Towards building an Inclusive Multilingual Speech Dataset for Indian Languages," 2024.

[21] S. B. Kalluri, P. Singh, P. R. Chowdhuri, A. Kulkarni, S. Baghel, P. Hegde, S. Sontakke, , D. K. T, S. R. M. Prasanna, D. Vijayasenan, and S. Ganapathy, "The Second DISPLACE Challenge : DIarization of SPeaker and LAnguage in Conversational Environments," in *Proc. INTERSPEECH 2024*, 2024.