BUT Systems for IWSLT 2023 Marathi - Hindi Low Resource Speech Translation Task

Santosh Kesiraju, Karel Beneš, Maksim Tikhonov and **Jan Černocký**
Speech@FIT, Brno University of Technology, Czechia
{kesiraju,ibenes,cernocky}@fit.vutbr.cz,
xtikho00@stud.fit.vutbr.cz

Abstract

This paper describes the systems submitted for Marathi to Hindi low-resource speech translation task. Our primary submission is based on an end-to-end direct speech translation system, whereas the contrastive one is a cascaded system. The backbone of both the systems is a Hindi-Marathi bilingual ASR system trained on 2790 hours of imperfect transcribed speech. The end-to-end speech translation system was directly initialized from the ASR, and then fine-tuned for direct speech translation with an auxiliary CTC loss for translation. The MT model for the cascaded system is initialized from a cross-lingual language model, which was then fine-tuned using 1.6 M parallel sentences. All our systems were trained from scratch on publicly available datasets. In the end, we use a language model to re-score the n-best hypotheses. Our primary submission achieved 30.5 and 39.6 BLEU whereas the contrastive system obtained 21.7 and 28.6 BLEU on official dev and test sets respectively. The paper also presents the analysis on several experiments that were conducted and outlines the strategies for improving speech translation in low-resource scenarios.

1 Introduction

A typical end-to-end (E2E) speech translation model is trained with the help of data triplets $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, i.e., the speech signal $(\boldsymbol{x})$ in source language, along with its transcription $(\boldsymbol{y})$, and, text translation $(\boldsymbol{z})$ in target language. In usual low-resource scenarios, the transcriptions in source language are unavailable and moreover the speech signal and the translation pairs $(\boldsymbol{x}, \boldsymbol{z})$ are also limited, which is the case for the IWSLT 2023 Marathi to Hindi low-resource speech translation task (Agarwal et al., 2023). In such cases, one can rely on transfer learning, where models trained on relatively large amounts of data (possibly on a related task such as automatic speech recognition) are transferred (adapted) to the target task/scenario (such as speech translation) using little amounts of labelled data (Bansal et al., 2019). To be specific, we train automatic speech recognition (ASR) systems on relatively large amount of transcribed speech data (2790 hours), and transfer the model for speech translation task by fine-tuning it on relatively small amount (16 hours) of IWSLT Marathi-Hindi training data.

This paper describes the systems submitted for the aforementioned task. While building the systems, we mainly focused on end-to-end systems, which resulted in our primary submission. We have also put some efforts in building a cascade pipeline that was submitted as a contrastive system. Both the systems come under the unconstrained category, i.e., we relied on external, publicly available data to train models. These models, which we refer to as *back-bone models*, mainly comprise automatic speech recognition (ASR), machine translation (MT) and language models (LM).

The Section 2 describes the various datasets used for training the back-bone models, and Section 3 presents the details of each individual back-bone models (ASR, MT, LM), followed by description of transfer learning for actual speech translation systems in Section 4. The Section 5 gives the results and analysis, quantifying the effect of various factors on the target translation task. Finally, we conclude in Section 6 and discuss directions for future works.

2 Datasets for training

Here we describe the details and present the statistics of various datasets used for training the back-bone models. These datasets come under various categories, i.e., paired speech data for training ASR, parallel text data for training MT and monolingual data for training LMs. All the data we considered for training covers only Hindi and Marathi languages. Both these share the same Devanagari script (unicode block) but there a few set of charac-

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 227–234

ters that are mutually exclusive.

2.1 Paired speech data

The paired speech data for Marathi and Hindi are collected from various publicly available datasets as listed below:

- **GramVaani (GV)**[1] comprises telephone quality speech in Hindi (hi). The dataset was used for Interspeech 2022 special session (Bhanushali et al., 2022; Patel and Scharenborg, 2022). We considered only the 100 hour labelled split of the dataset.
- **Indian Language Corpora (ILC)** (Abraham et al., 2020)[2] is crowdsourced speech data along with transcriptions in Marathi language. The dataset is collected from 36 participants with various socio-economic backgrounds and dialects.
- **Mozilla Common Voice v12 (MCV)** (Ardila et al., 2020) is a crowdsource collection of paired speech data across various languages. We took the validated versions of Hindi (hi) and Marathi (mr) from this corpus.
- **MUCS** (Diwan et al., 2021)[3] is multilingual and code-switched corpus for training ASR systems in 6 different Indian languages. The dataset was introduced in Interspeech 2021 as part of a special session focusing on ASR for Indian languages. We considered Hindi and Marathi data from this corpus. Although MUCS contains about 100 hours of transcribed speech for both Marathi and Hindi, the lexical content is not diverse, i.e., the same utterances were spoken by various speakers.
- **Multi-speaker speech corpora (MSSC)** (He et al., 2020)[4] is a collection of clean speech data with transcriptions intended for building text-to-speech synthesis systems for various Indian languages. We considered only the Marathi split from this corpus.
- **Shrutilipi (SL)**[5] is collected from public archives and contains about 6400 hours of radio broadcast news in various Indian languages. The corresponding transcriptions were obtained with the help of OCR and other heuristics (Bhogale et al., 2022). This corpus is the bigger chunk of the data we used for training, but the transcriptions obtained are not accurate. A manual inspection revealed some erroneous alignments at the beginning and end of the utterances. By setting a threshold (≥ 85) on the provided alignment score, we filtered Hindi (hi) and Marathi (mr) data from this corpus. We believe the domain of this data is closer to IWSLT 2023 speech translation data.

The statistics of each of the above datasets is presented in Table 1. This data was used to train mono and bilingual ASR systems that are described later in Section 3.1. All the speech data was upsampled to 16 kHz. Using Kaldi toolkit (Povey et al., 2011) 80 dimensional filter banks and 3-dimensional pitch features are extracted for every 25 ms of speech frame sliding with 10 ms.

2.2 Monolingual and parallel text data

We prepared monolingual data for both Hindi and Marathi. We pooled data from transcribed speech (Table 1), Samanantar (Ramesh et al., 2022), Indic2Indic, IIIT-H CVIT (Siripragada et al., 2020) corpus, resulting in 9 M sentences (217 M tokens) for Hindi and 4M sentences for Marathi[6].

The parallel text was taken only from Indic2Indic split from Samanantar (Ramesh et al., 2022), whose statistics are given in Table 2. We retained punctuation in all the text.

2.3 Speech translation data

The official speech translation data for Marathi - Hindi involves around 16 hours of training split, i.e., Marathi speech and its translations in Hindi. There are no transcriptions for the Marathi speech. Table 3 presents the statistics of the provided speech translation data. We used speed perturbation (0.9, 1.0, 1.1) to augment the speech translation data. The effect of such augmentation on the final translation performance is discussed later in Section 5.

[1] https://sites.google.com/view/gramvaaniasrchallenge/

[2] https://www.cse.iitb.ac.in/~pjyothi/indiccorpora/

[3] https://navana-tech.github.io/MUCS2021/data.html

[4] https://www.openslr.org/64/

[5] https://ai4bharat.org/shrutilipi

[6] Due to a bug in data preparation, only Shrutilipi text data 400 K (8.2 M tokens) out of 4 M sentences were used to train Marathi LM.

Dataset	Language	Duration in hours (number of utterances) Training		Dev		Test	
GV	hi	97.9	(37,152)	4.9	(1885)	2.8	(1032)
ILC	mr	109.2	(92,471)	-	-	-	-
MCV	hi	5.3	(4481)	2.8	(2179)	4.1	(2962)
	mr	12.0	(7321)	3.0	(1678)	3.2	(1827)
MUCS	hi	95.1	(99,925)	5.6	(3843)	-	-
	mr	93.8	(79,432)	5.0	(4675)	-	-
MSSC	mr	3.0	(1569)	-	-	-	-
SL	hi	1478.6	(764,237)	-	-	-	-
	mr	894.8	(466,203)	-	-	-	-
Total	hi	1676.8	(898,369)	13.3	(7895)	6.9	(3994)
	mr	1112.8	(638,159)	8.0	(6353)	3.2	(1827)

Table 1: Statistics of the data used for training ASR systems. The dev and test splits are only used for internal evaluation of the ASR systems.

Number of utterance pairs Training	Dev	Test
1634551	2000	2000

Table 2: Number of parallel utterance (sentence) pairs between Marathi-Hindi that are used for training XLM and MT models.

3 Back-bone models

Here, we describe the architecture and training details of various backbone models.

3.1 ASR

The ASR model is a transformer based seq2seq model. The speech features are passed through 2 layers of convolution, followed by 12 layers of transformer encoder blocks and 6 layer of transformer decoder blocks, with $d_{\text{model}} = \{256, 512\}$[7], heads $= 4$, $d_{\text{ff}} = 2048$. The dropout was set to 0.1. The model is trained with a batch size of 128 for 100 epochs using Adam optimizer (Kingma and Ba, 2015), and warm up scheduler with a peak learning rate of 0.0005. The training is done with joint CTC and attention objective (Karita et al., 2019), where the CTC is applied at the end of encoder layer and the attention acts at the output of autoregressive decoder (teacher-forcing).

$$\mathcal{L}_{\text{asr}} = \alpha\, \mathcal{L}_{\text{ctc}}(\boldsymbol{x}, \boldsymbol{y}) + (1 - \alpha)\mathcal{L}_{\text{att}}(\boldsymbol{x}, \boldsymbol{y}). \quad (1)$$

In case of bilingual ASR, the CTC layer, input and output layers of the decoder are specific to each language, i.e., the (sub-)word embeddings are *not* shared across languages. Such a design ensures that only target language tokens are decoded, irrespective of the phonetic similarity with other languages in the model. The ASR models were trained using ESPnet toolkit (Watanabe et al., 2018). The performance of various mono and bilingual ASR systems is discussed later in Section 5.

3.2 XLM

The architecture of pre-training masked-language model is based on cross-lingual language model (XLM) (Lample and Conneau, 2019)[8]. More specifically, we use translation language modelling objective along with masked language modelling to train the transformer based encoder. Here, we use BPE-based sub-word vocabulary that is obtained jointly for both languages. The model has 6 transformer blocks with 512 embedding dimension, 8 attention heads, dropout of 0.1 for both attention and feed-forward layers. The model is trained for a maximum of 1000 epochs using Adam optimizer with a learning rate of 0.0001.

[7] Smaller models use $d_{\text{model}} = 256$, where as bigger models use $d_{\text{model}} = 512$.

[8] https://github.com/facebookresearch/XLM

Duration in hours (# utterances)		
Training	Dev	Test
15.9 (7990)	3.7 (2103)	4.4 (2164)

Table 3: Statistics of Marathi-Hindi IWSLT2023 speech translation data.

3.3 MT

The MT model is a transformer based seq2seq model initialized from XLM. Both the encoder and decoder parameters are initialized from XLM encoder, except for the cross-attention parameters in the decoder that are randomly initialized. The model is then fine-tuned on the same 1.6 M parallel sentences with a batch size of 64 and a maximum of 1000 epochs. The model achieved **23.0** and **22.6** BLEU scores on the internal valid and test sets (Table 2) respectively.

3.4 LM for re-scoring

For Hindi, we used an LSTM of three layers of 4096 units each, with no dropout. The model was trained on 217 M sub-word tokens obtained by tokenizing the monolingual Hindi corpus into a 10k Unigram vocabulary (Kudo, 2018). The model achieved validation perplexity of 46. Thereafter, we have fine-tuned it on text data from Shrutilipi (SL) data for 500 steps.

For Marathi, we used an LSTM of 2 layers per 2048 units, again with no dropout. This model also utilized a 10k Unigram vocabulary and was trained on 8.2 M tokens. This model achieved validation perplexity of 120.

4 Speech translation systems

Here, we briefly describe both the end-to-end and cascade systems.

4.1 End-to-end

The E2E models are initialized from pre-trained ASR models. We use both the encoder and decoder from the ASR, as it provides a better initialization since the representations from the encoder are readily compatible with the decoder (Bansal et al., 2019). The model is then trained for direct speech translation, with the auxiliary CTC objective also for translation (Zhang et al., 2022; Yan et al., 2023; Kesiraju et al., 2023).

$$\mathcal{L}_{\text{st}} = \lambda\, \mathcal{L}_{\text{ctc}}(\boldsymbol{x}, \boldsymbol{z}) + (1-\lambda)\mathcal{L}_{\text{att}}(\boldsymbol{x}, \boldsymbol{z}) \quad (2)$$

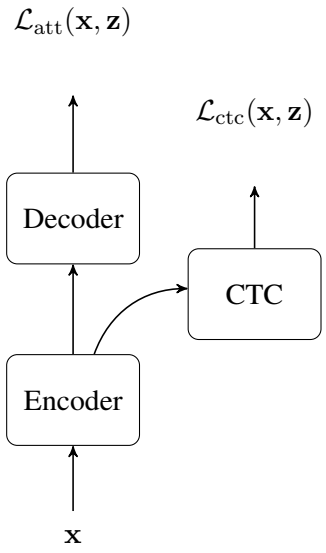

Figure 1: End-to-end framework for speech translation. $\mathbf{x}$ is the input speech (features), $\mathbf{z}$ is the target text translation.

The effect of various initializations and their influence on downstream speech translation is discussed later in Section 5.

The E2E speech translation was also trained using ESPnet toolkit. Our changes to the original toolkit, along with the training recipes, are available online[9].

A beam search based joint decoding (Karita et al., 2019) that relies on the weighted average of log-likelihoods from both the CTC and transformer decoder modules is used, that produces the most likely hypotheses according to

$$\hat{\mathbf{z}} = \arg\max_{\mathbf{z}} \; \beta \, \log p_{\text{ctc}}(\mathbf{z} \mid \mathbf{x}) + (1-\beta) \, \log p_{\text{att}}(\mathbf{z} \mid \mathbf{x}) \quad (3)$$

We found $\lambda = \{0.1, 0.3\}$, $\beta = \{0.1, 0.3\}$ suitable for joint training and decoding respectively.

4.2 Cascade systems

For the cascade speech translation systems, we first decode n-best hypotheses from ASR model and obtain 1-best from Marathi LM rescorer. These are then passed directly to the MT system, which gives us n-best translation hypotheses in target language Hindi. These are then re-scored by Hindi LM to give us 1-best translation hypotheses.

[9] https://github.com/BUTSpeechFIT/espnet/tree/main/egs2/iwslt23_low_resource/st1

Model name	Training data (hrs)	Model type	Sub-word vocab per language	Dev WER		Test WER	
				mr	hi	mr	hi
H1	198†	Mono (hi)	1000	-	30.7	-	35.9
H2	1676	Mono (hi)	8000	-	24.7	-	28.4
M1	218†	Mono (mr)	1000	14.3	-	42.4	-
M2	1112	Mono (mr)	8000	19.0	-	36.0	-
B1	416†	Bilingual (mr, hi)	1000	11.1	31.5	31.9	35.1
B2	2789	Bilingual (mr, hi)	8000	16.0	24.2	23.7	26.9

Table 4: Word-error-rates (WER) of various mono and bilingual ASR systems, trained on various amounts of data. † implies that the training data contains everything from Table 1 except Shrutilipi (SL).

A further fine-tuning of the MT system using 1-best hypotheses from Marathi to Hindi IWSLT training set did not improve the results. Due to time constraints, we did not try various strategies (Bentivogli et al., 2021) or hyperparameter tuning for the cascade systems.

4.3 Re-scoring n-best hypotheses

We have utilized the language models to re-score up to 100-best hypotheses in both languages. Using BrnoLM[10], we have introduced the language model scores. Here, we have tuned the two hyperparameters: The weight of the LM score (additive to 1.0 weight of the acoustic system) and an insertion bonus, added for each token of the hypothesis, in the LM tokenization. For the E2E system, we have achieved optimal results with LM weight 1.2 and insertion bonus 5.5. For the Marathi ASR in the cascade system, optimal setting was 0.3 and 3.5. For the translation system in the cascade, we did not achieve any improvement by re-scoring the output with the Hindi LM.

5 Results and analysis

Here, we present the performance of various backbone models, along with analysis showing the effectiveness of various factors such as initializations, data augmentation, auxiliary objectives and joint decoding.

5.1 Performance of ASR systems

From the Table 4 we can see that the bilingual models perform (B1, B2) better than the monolingual parts (H1, M1, H2, M2). Here, H1, M1 and B1 are smaller models with $d_{\text{model}} = 256$, whereas H2, M2 and B2 are bigger ones with $d_{\text{model}} = 512$. All the ASR models were trained with joint CTC and attention loss, where the CTC weight of 0.3 was found to be optimal. The same weight was used during joint decoding. Since we retained the original punctuation in the text, the WER is slightly affected.

5.2 Performance of ST

Here we present the results of speech translation systems based on end-to-end architecture. As shown in Table 5, all the ST models were initialized either from mono or bilingual ASR systems and fine-tuned using the speech translation data (with or without data augmentation). While most of these systems can be considered direct end-to-end; using an external LM for re-scoring the n-best makes an exception. Using a Marathi monolingual ASR model would be sub optimal because the internal language model represented in the decoder of the ASR would not be suitable for generating linguistically acceptable text sequences in Hindi.

Fig. 2 shows the effect of CTC weight during joint training and decoding. We can see that 0.3 is the optimal weight both for training and decoding. Since, we have a separate vocabulary for both the languages, the posterior probabilities from CTC during joint decoding will only correspond to the tokens from the target language Hindi. This is important, since both the languages come from same family with high phonetic similarity, and use same Devanagari script, the non auto regressive CTC decoder does not accidentally provide higher scores for tokens from source language Marathi. The latter scenario can happen when using a joint-sub word vocabulary for both the languages.

Sacrebleu library (Post, 2018) was used to com-

[10] https://github.com/BUTSpeechFIT/BrnoLM

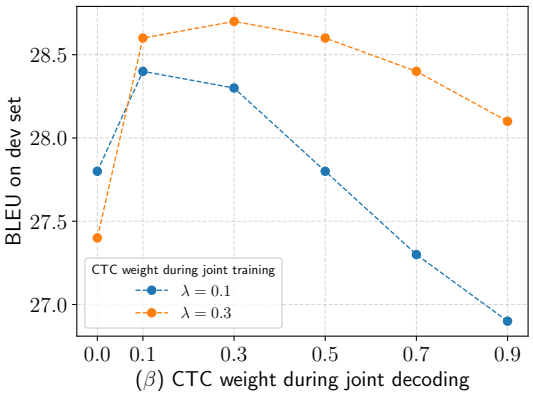

Figure 2: Effect of hyperparameters in joint training and decoding for direct speech translation. The model is initialized from B2 and trained on augmented training data.

pute BLEU[11] and CHRF2[12] scores in the dev sets.

From the Table 5, we can see that independent improvements come from using bilingual ASR trained on more data, data augmentation (speed perturbation) and LM re-scoring. In case of cascade system, the LM re-scoring did not improve the results. We believe this is because the Marathi LM was trained on much fewer amounts of data (400K sentences). We plan to rerun these experiments in the near future.

Finally, our primary submission was based on B2 + ST fine-tuning with data augmentation + LM re-scoring which obtained **39.6** BLEU and **63.3** CHRF2 scores on official test set. Our contrastive system was based on B2 + MT + LM re-scoring which obtained **28.6** BLEU and **54.4** CHRF2 scores.

A manual inspection of the translation outputs revealed that several mismatches occurred where there are ambiguous numerals, i.e., some numbers were written using digits while the others were spelled out verbatim. There are also cases where both notations were mixed. We believe, further text normalization of both reference and hypothesis could give us a better picture of the evaluation scores.

[11] `nrefs:1 | case:mixed | eff:no | tok:13a | smooth:exp | version:2.3.1`

[12] `nrefs:1 | case:mixed | eff:yes | nc:6 | nw:0 | space:no | version:2.3.1`

ST Model initialization	Speed perturb	Dev set BLEU	Dev set CHRF2
H1	✗	16.3	45.0
H2	✓	24.9	51.0
B1	✗	17.4	46.2
B1	✓	20.1	48.2
B2	✓	28.7	54.4
B2 + LM rescore	✓	30.6	55.9
Cascade	-	21.7	48.2

Table 5: Speech translation results on Marathi - Hindi dev set. All the ST models are fine-tuned on training data from Table 3.

6 Conclusions

In this paper, we presented the systems submitted to the IWSLT 2023 Marathi Hindi low resource track. Our main efforts were along the end-to-end direct speech translation system, initialized from a bilingual ASR. The model was jointly trained with CTC and attention objective directly for translation. The joint decoding provided additional benefits. These strategies combined with speed perturbation for data augmentation and re-scoring the n-best hypotheses using external LM provided further significant improvements. We also submitted a cascade system which uses the same bilingual ASR as the backbone, followed by an MT system. Both systems performed competitively, while the one based on end-to-end provided superior results in terms of BLEU. It is yet to be investigated, if the large pre-trained MT systems would close the gap between cascade and end-to-end systems.

Acknowledgements

The work was supported by Czech National Science Foundation (GACR) project NEUREM3 No. 19-26934X, Czech Ministry of Education, Youth and Sports project no. LTAIN19087 "Multi-linguality in speech technologies" and Horizon 2020 Marie Skłodowska-Curie grant ESPERANTO, No. 101007666. Computing on IT4I supercomputer was supported by the Czech Ministry of Education, Youth and Sports through the e-INFRA CZ (ID:90254). We thank the reviewers for their constructive feedback.

References

Basil Abraham, Danish Goel, Divya Siddarth, Kalika Bali, Manu Chopra, Monojit Choudhury, Pratik Joshi, Preethi Jyoti, Sunayana Sitaram, and Vivek Seshadri. 2020. Crowdsourcing speech data for low-resource languages from low-income workers. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2819–2826, Marseille, France. European Language Resources Association.

Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Yannick Estève, Kevin Duh, Marcello Federico, Souhir Gahbiche, Barry Haddow, Benjamin Hsu, Phu Mon Htut, Hirofumi Inaguma, Dávid Javorský, John Judge, Yasumasa Kano, Tom Ko, Rishu Kumar, Pengwei Li, Xutail Ma, Prashant Mathur, Evgeny Matusov, Paul McNamee, John P. McCrae, Kenton Murray, Maria Nadejde, Satoshi Nakamura, Matteo Negri, Ha Nguyen, Jan Niehues, Xing Niu, Atul Ojha Kr., John E. Ortega, Proyag Pal, Juan Pino, Lonneke van der Plas, Peter Polák, Elijah Rippeth, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Yun Tang, Brian Thompson, Kevin Tran, Marco Turchi, Alex Waibel, Mingxuan Wang, Shinji Watanabe, and Rodolfo Zevallos. 2023. Findings of the IWSLT 2023 Evaluation Campaign. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*. Association for Computational Linguistics.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Sameer Bansal, Herman Kamper, Karen Livescu, Adam Lopez, and Sharon Goldwater. 2019. Pre-training on high-resource speech recognition improves low-resource speech-to-text translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 58–68, Minneapolis, Minnesota. Association for Computational Linguistics.

Luisa Bentivogli, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus direct speech translation: Do the differences still make a difference? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2873–2887, Online. Association for Computational Linguistics.

Anish Bhanushali, Grant Bridgman, Deekshitha G, Prasanta Ghosh, Pratik Kumar, Saurabh Kumar, Adithya Raj Kolladath, Nithya Ravi, Aaditeshwar Seth, Ashish Seth, Abhayjeet Singh, Vrunda Sukhadia, Umesh S, Sathvik Udupa, and Lodagala V. S. V. Durga Prasad. 2022. Gram Vaani ASR Challenge on spontaneous telephone speech recordings in regional variations of Hindi. In *Proc. Interspeech 2022*, pages 3548–3552.

Kaushal Santosh Bhogale, Abhigyan Raman, Tahir Javed, Sumanth Doddapaneni, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. 2022. Effectiveness of mining audio and text pairs from public data for improving asr systems for low-resource languages.

Anuj Diwan, Rakesh Vaideeswaran, Sanket Shah, Ankita Singh, Srinivasa Raghavan, Shreya Khare, Vinit Unni, Saurabh Vyas, Akash Rajpuria, Chiranjeevi Yarra, Ashish Mittal, Prasanta Kumar Ghosh, Preethi Jyothi, Kalika Bali, Vivek Seshadri, Sunayana Sitaram, Samarth Bharadwaj, Jai Nanavati, Raoul Nanavati, and Karthik Sankaranarayanan. 2021. MUCS 2021: Multilingual and Code-Switching ASR Challenges for Low Resource Indian Languages. In *Proc. Interspeech 2021*, pages 2446–2450.

Fei He, Shan-Hui Cathy Chu, Oddur Kjartansson, Clara Rivera, Anna Katanova, Alexander Gutkin, Isin Demirsahin, Cibu Johny, Martin Jansche, Supheakmungkol Sarin, and Knot Pipatsrisawat. 2020. Open-source multi-speaker speech corpora for building Gujarati, Kannada, Malayalam, Marathi, Tamil and Telugu speech synthesis systems. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6494–6503, Marseille, France. European Language Resources Association.

Shigeki Karita, Nelson Enrique Yalta Soplin, Shinji Watanabe, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani. 2019. Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration. In *Proc. of Interspeech*, pages 1408–1412.

Santosh Kesiraju, Marek Sarvaš, Tomáš Pavliček, Cécile Macaire, and Alejandro Ciuba. 2023. Strategies for improving low resource speech to text translation relying on pre-trained asr models.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.

Tanvina Patel and Odette Scharenborg. 2022. Using cross-model learnings for the Gram Vaani ASR Challenge 2022. In *Proc. Interspeech 2022*, pages 4880–4884.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukáš Burget, Ondřej Glembek, K. Nagendra Goel, Mirko Hannemann, Petr Motlíček, Yanmin Qian, Petr Schwarz, Jan Silovský, Georg Stemmer, and Karel Veselý. 2011. The kaldi speech recognition toolkit. In *Proceedings of ASRU 2011*, pages 1–4. IEEE Signal Processing Society.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Shashank Siripragada, Jerin Philip, Vinay P. Namboodiri, and C V Jawahar. 2020. A multilingual parallel corpora collection effort for Indian languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3743–3751, Marseille, France. European Language Resources Association.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. ESPnet: End-to-end speech processing toolkit. In *Proceedings of Interspeech*, pages 2207–2211.

Brian Yan, Siddharth Dalmia, Yosuke Higuchi, Graham Neubig, Florian Metze, Alan W Black, and Shinji Watanabe. 2023. CTC alignments improve autoregressive translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1623–1639, Dubrovnik, Croatia. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Rico Sennrich. 2022. Revisiting End-to-End Speech-to-Text Translation From Scratch. In *International Conference on Machine Learning*, volume 162 of *Proc. of Machine Learning Research*, pages 26193–26205. PMLR.