Improving Low Resource Speech Translation with Data Augmentation and Ensemble Strategies

Akshaya Vishnu Kudlu Shanbhogue **Ran Xue*** **Soumya Saha***
Daniel Yue Zhang* **Ashwinkumar Ganesan***
Amazon Alexa AI
{ashanbho,ranxue,soumyasa,dyz,gashwink}@amazon.com

Abstract

This paper describes the speech translation system submitted as part of the IWSLT 2023 shared task on low resource speech translation. The low resource task aids in building models for language pairs where the training corpus is limited. In this paper, we focus on two language pairs, namely, Tamasheq-French (Tmh→Fra) and Marathi-Hindi (Mr→Hi) and implement a speech translation system that is *unconstrained*. We evaluate three strategies in our system: (a) Data augmentation where we perform different operations on audio as well as text samples, (b) an ensemble model that integrates a set of models trained using a combination of augmentation strategies, and (c) post-processing techniques where we explore the use of large language models (LLMs) to improve the quality of sentences that are generated. Experiments show how data augmentation can relatively improve the BLEU score by 5.2% over the baseline system for Tmh→Fra while an ensemble model further improves performance by 17% for Tmh→Fra and 23% for Mr→Hi task.

1 Introduction

Speech translation (ST) systems have multiple applications. They can be utilized in a wide range of scenarios such as closed captioning in different languages while watching videos or even as a real-time assistant that translates speeches to live audiences. One persistent challenge for speech translation systems continues to be performing translations for low resource language pairs.[1] The IWSLT 2023 (Agarwal et al., 2023) shared task for low resource speech translation targets 8 language pairs that include Tunisian Arabic (Aeg) to English (En), Irish (Ga) to English (En), Marathi (Mr) to Hindi (Hi), Maltese (Mlt) to English (En), Pashto (Pus) to French (Fr), Tamasheq (Tmh) to French (Fr), and Quechua (Que) to Spanish (Es). This paper outlines a low resource speech translation system (from the Amazon Alexa AI team) for 2 language pairs, namely, Tamasheq-French (**Tmh→Fra**) and Marathi-Hindi (**Mr→Hi**).

Depending on the type of output that is generated the end-to-end speech translation task has two formats: (a) Speech-to-text (S2T), and (b) Speech-to-Speech (S2S). There are two types of ST systems. The first is a cascaded system where speech recognition and language translation are decoupled.[2] The second is an end-to-end (E2E) model that combines both audio processing and language translation. We design and evaluate an E2E model in this paper.

In the past, various approaches have been proposed to build E2E low resource speech translation models. Bansal et al. (2018) designs an initial system that is an encoder-decoder architecture that integrates a convolutional neural network (CNN) and recurrent neural network (RNN). Stoian et al. (2020) try to improve ST models for low resource languages by pretraininig the model on automated speech recognition (ASR) task. Cheng et al. (2021) propose a new learning framework called AlloST that trains a transformer architecture with language-independent phonemes. Mi et al. (2022) improves translation performance by expanding the training corpus through generation of synthetic translation examples, where the target sequences are replaced with diverse paraphrases. In IWSLT 2022 (Anastasopoulos et al., 2022), Boito et al. (2022b) utilized a wav2vec encoder and trained an E2E ST model where source audios are directly translated to the target language.[3]

In this paper, we extend the previous work with the following contributions:

- We train and assess a speech translation model

* These authors contributed equally to this work.
[1] https://iwslt.org/2023/low-resource

[2] For the S2S version, speech generation is separate too.
[3] They contributed towards the low resource speech translation task for Tmh→Fra.

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 241–250
July 13-14, 2023 ©2023 Association for Computational Linguistics

for Tmh→Fra with audio stretching (Yang et al., 2021).

- The baseline model for Tmh→Fra is trained with a back-translation corpus generated using the NLLB-200 machine translation model (Team et al., 2022).
- For Tmh→Fra, we build a separate training corpus of paraphrases and show that model performance improves when trained on this dataset (Bhavsar et al., 2022).
- We show how a weighted cross entropy loss further improves the performance of the Tmh→Fra translation model. The model trained with this loss, additional data generated using paraphrases and audio stretching is shown to perform 5.2% better than the baseline.
- An ensemble of models trained on the above strategies shows the best performance, with BLEU score that is 17.2% higher than the average BLEU score of the individual models within the ensemble.
- In case of Mr→ Hi, our best independent ensemble model shows a 23% improvement over the average BLEU score of the individual models within the ensemble.

Apart from these contributions, we also explore post-processing techniques with large language models (LLMs), focusing on re-ranking generated translations (Kannan et al., 2018), correcting the grammar of translations and masking tokens so that the LLM can complete the translate sentence. These methods though, did not yield any noticeable improvement.

The paper is organized as follows: Section 2 describes our speech translation system, 3.1 has details about the datasets for various language pairs, 3.2 contains analysis of our experimental results and we finally conclude in 4.

2 Speech Translation System

2.1 Baseline Model

Our base model for Tmh→Fra ST task is an end-to-end speech translation system which employs an encoder-decoder architecture (Vaswani et al., 2017). We initialize the audio feature extractor and the 6-layer transformer encoder from a pretrained wav2vec 2.0 base model (Baevski et al., 2020). We reuse the wav2vec 2.0 model pretrained on 243 hours of Tamasheq audio data released by ON-TRAC Consortium Systems (Boito et al., 2022b). During initialization, the last 6 layers of the pre-trained wav2vec 2.0 model are discarded. We use a shallow decoder which consists 2 transformer layers with 4 attention heads. Between encoder and decoder, we use one feed-forward layer to match the dimension of encoder output and decoder input.

During training, the model directly performs speech to text translation task without generating intermediate source language text. The training loss is the cross entropy loss between ground truth and hypothesis with label smoothing of 0.1. Each experiment is trained for 200 epochs and checkpoints are selected based on best validation BLEU.

For Marathi-Hindi speech-to-text (ST) model, we chose a Wav2Vec 2.0 base model finetuned on 960 h of English speech (Baevski et al., 2020) as the encoder baseline. We also used the same encoder model finetuned on 94 hours of Marathi audio data (Chadha et al., 2022) in our experiments. For these models, the last 6 layers of the pretrained models were discarded, while the decoder architecture and other hyperparameters were kept same as the Tmh→Fra models [4]. For audio encoder, we also experimented with Wav2vec 2.0 XLS-R 0.3B model (Babu et al., 2021) and another XLS-R 0.3B model specifically finetuned on Marathi audio (Bhattacharjee, 2022). Because the XLS-R base model was trained on audio from a range of Indian languages including Marathi and Hindi, we chose to incorporate XLS-R in our experimentation. For the XLS-R based models, we utilized the first 12 out of 24 encoder layers to initialize the encoder followed by a linear projection layer to transform the output features of 1024 dimensions to the desired decoder dimensionality of 256. We trained all Marathi-Hindi ST models for 300 epochs and we chose the best checkpoint based on validation BLEU score.

2.2 Data Augmentation

2.2.1 Audio Stretching

We apply audio stretching directly on wav form data using torchaudio library (Yang et al., 2021).[5] For each audio sample, we alter the speed of the audio with a rate uniformly sampled from $[0.8, 1.2]$ with a probability of 0.8 while maintaining the audio sample rate.

[4] Detailed hyperparameters used can be found in A.1.
[5] https://github.com/pytorch/audio

2.2.2 Back-Translation

We use the NLLB-200 machine translation model to generate variations of target text in French (Team et al., 2022). The original French data is first translated into English, and then translated back into French. For French to English translation, only 1 best prediction is used. For English to French translation, we take the top 5 results with a beam size of 5.

We also try to generate synthetic transcription of the Tamasheq audio by translating French text into Tamasheq. However, we notice that the translation quality is unstable and decide to not use it for the experiment.

2.2.3 Paraphrasing

We use a French paraphrase model (Bhavsar, 2022), which is a fine tuned version of mBART model (Liu et al., 2020), to generate variations of target text in French. We take the top 5 paraphrases using beam search with a beam size of 5.

2.2.4 Weighted Loss

As the quality of synthetically generated sentences varies, we apply a sentence level weight to the corresponding sample's cross entropy loss during training.

$$l = \sum_{i}^{N} w_i * CE(y_i, \hat{y}_i) \tag{1}$$

where N is the size of the corpus, y_i, $\hat{y}_i$, w_i are ground truth, prediction, and loss weight for sample i respectively . For back-translation data, the weights are directly taken from the prediction score of NLLB-200. For paraphrasing data, we calculate the perplexity of each generated paraphrase and then take the exponential of the perplexity as the weight. For original training data (clean and full), weight are set to 1.

2.3 Ensemble Model

Ensemble decoding (Liu et al., 2018; Zhang and Ao, 2022) is a method of combining probability values generated by multiple models while decoding the next token. We provide equal-weight to N different ensemble models as shown in 2.

$$logP(y_t|x, y_{1...t-1}) = \frac{1}{N}\sum_{i}^{N} logP_{\theta_i}(y_t|x, y_{1...t-1}) \tag{2}$$

Where, y_t denotes the decoded token at time t, x denotes the input and θ_i denotes the ith model in the ensemble.

We apply the following ensemble decoding strategies:

- Independent ensemble: we ensemble checkpoints having the highest BLEU scores on the validation set, on N training runs. The N different models have the same architecture, but initialized with different seed values.
- Data-augmented ensemble: we ensemble checkpoints having the highest BLEU scores on the validation set, on N training runs. The N different models have the same architecture, but trained on different data augmentation strategies.

We additionally attempt a checkpoint ensemble, where N different checkpoints having the highest validation BLEU within the same training run are ensembled. Since we notice marginal improvements with checkpoint ensemble, we decide to not explore checkpoint ensemble in depth for our experiments.

2.4 Post Processing with LLMs

We further explore a set of post processing strategies by leveraging large language models (LLM) to 1) rerank the top-k generated samples; 2) correct grammar of the output; and 3) guess the missing tokens of the sentence. The strategy is based on the observation that translation outputs from the validation set often carry incomplete sentences and broken grammar. We found that LLMs are good fit to address this problem as they have brought promising improvements in sentence re-ranking, and rewriting tasks (Liu et al., 2023). We summarize our proposed strategies as follows:

2.4.1 Re-ranking

The reranking approach takes the top 5 results from the best-performing candidate, and rerank these outputs with language models. We first explore performing shallow fusion (Kannan et al., 2018) with language model (GPT2-Fr).[6] Additionally, we leverage a LLM (French finetuned-Alpaca 7B [7]) to guess the most probable sentence that is from a radio broadcast news with the prompt:

quelle phrase est plus susceptible d'apparaître dans un journal télévisé

[6]https://github.com/aquadzn/gpt2-french

[7]https://github.com/bofenghuang/vigogne

2.4.2 Sentence Correction

The sentence correction approach rewrites the whole output prediction by correcting the grammatical and spelling errors. We use two LLMs for this tasks - aforementioned Alpaca model and Bloom 7B with the following prompt: [8]

> *Corrigez la faute de frappe et la grammaire de la phrase sans changer la structure*

2.4.3 Token Masking

The token masking approach first masks the translation output with <blank> tokens for out-of-vocabulary (OOV) tokens. For example the predicted output "...Les questions sont [pi];." is replaced with " <blank> Les questions sont <blank>." where [pi] is a common token we observed in the prediction output that does not carry meaning. We then apply the following prompt to let the LLMs to complete the sentence:

> *complétez la phrase en remplaçant les jetons <blank>*

3 Experiments

3.1 Datasets

3.1.1 Tamasheq-French Corpus

The dataset used for our training, validation, and testing is obtained from Boito et al. (2022a), which is shared as a part of IWSLT 2023 shared task. It consists of a parallel corpus of radio recordings in Tamasheq language predominantly from male speakers. The dataset includes approximately 18 hours of speech divided in training, validation and test sets along with its French translation. We refer to this data as "clean". Additionally, there is approximately 2 hours of possible noisy training data from the same source, which we include in our experiments along with the clean data. We refer to this combined 20 hour dataset as "full" data. The statistics of the dataset are in Table 2.

Data Split	Hours	# Utterances
train clean	13.6	4,444
train full	15.5	4,886
valid	1.7	581
test2022	2	804
test2023	1	374

Table 2: **Data statistics for tmh→fra corpus.** *Hours* shows the number of hours of audio samples available while *# Utterances* is the associated number of utterances.

[8]https://huggingface.co/bigscience/bloom-7b1

3.1.2 Marathi-Hindi Corpus

For Marathi-Hindi we use the data from Panlingua (2023) containing approximately 25 hours of speech. The audio recordings are sourced from the news domain. The statistics of the dataset is shown in Table 3.

Data Split	Hours	# Utterances
train	16	7,990
valid	3.7	2,103
test	4.5	2,164

Table 3: **Data statistics for mr→hi corpus.** *Hours* shows the number of hours of audio samples available while *# Utterances* is the associated number of utterances.

3.2 Experimental Results

In this section, we compare the effects of data augmentation, ensembling and post-processing strategies on the tmh→fra task on test 2022 dataset. We additionally compare results on the mr→hi task on the validation dataset.

3.2.1 Impact of Data Augmentation

Table 1 shows the effect of various data augmentation strategies used. We find that using full-audio dataset performs better than using just the clean-audio data. Also, adding audio stretching alone does not improve model performance.

Adding synthetically generated back-translation data shows mixed results. We hypothesize that this is due to cascading errors while performing back-translation. However, adding paraphrases data performs slightly better than baseline. We find that using a weighted loss while using synthetically generated translation data is beneficial.

3.2.2 Performance of Ensemble Model

Table 6 shows the summary of the effect of different ensembling strategies. For complete results, refer to table 12. We find that the performance of the ensemble model increases with the increase in number of models present in the ensemble. We also find that the data-augmented ensemble works better than independent ensemble. Additionally, data-augmented ensembling using paraphrase data performs better than data-augmented ensembling using back-translation data.

3.2.3 Impact of Post-processing Methods

Table 4 summarizes the experimental results for the post-editing strategies. We make the following observations. First, sentence correction strategy

#	Data	Data Augmentation	Vocab size	Loss	Test2022 BLEU
cb	clean	baseline	1k	baseline	8.85
fb	full	baseline	1k	baseline	9.25
ft	full	back-translation	3k	baseline	8.84
ftw	full	back-translation	3k	weighted	9.45
fta	full	back-translation + audio stretching	3k	baseline	9.01
ftaw	full	back-translation + audio stretching	3k	weighted	**9.71**
fp	full	paraphrase	3k	baseline	**9.70**
fpw	full	paraphrase	3k	weighted	**9.73**
fpa	full	paraphrase + audio stretching	3k	baseline	9.47
fpaw	full	paraphrase + audio stretching	3k	weighted	9.53

Table 1: **Impact of Data Augmentation on tmh→fra models.** The table shows the BLEU scores for different strategies in comparison to the baseline trained on *clean* and *full* dataset. *Back-Translation + audio stretching* and *Paraphrase* dataset augmentation improve the BLEU score. *Back-Translation* alone can improve model performance when combined with a weighted loss.

Approach	Model	Test2022 BLEU
Baseline	Ensembled Wav2Vec2	11.26
Reranking	Shallow-Fusion-based (GPT2-French)	11.24
	Instruct-based (Stanford Alpaca 7B)	10.78
Token Masking	Stanford Alpaca 7B	11.20
	Bloom 6.7B	10.84
Sentence Correction	Stanford Alpaca 7B	8.70
	Bloom 6.7B	8.54
Translation + Reranking	Stanford Alpaca 7B	3.45
	Bloom 6.7B	3.58

Table 4: **Impact of Post Processing on tmh→fra corpus.** The post-processing steps outlined are applied to an *Ensembled Wav2Vec2* model. The post-processing with a LLM does not provide any additional benefit.

Reranking	*Instruct*: quelle phrase est plus susceptible d'apparaître dans un journal télévisé *Input*: top k hypothesis *Output*: best hypothesis picked by LLM
Token Masking	*Instruct*: complétez la phrase en remplaçant les jetons <blank>? *Input*: Donc, on dirait que l'organisation de l'UENA, elle est <blank> *Output*: Donc, on dirait que l'organisation de l'UENA, elle est **un organisme de bienfaits**
Sentence Correction	*Instruct*: Corrigez la faute de frappe et la grammaire de la phrase sans changer la structure *Input*: Les a été libérés et ceux qui sont rentrés. *Output*: Ils ont été libéré et ceux rentrant.

Table 5: **Prompt Designs.** Example LLM Prompts for Post Processing tmh→fra corpus.

Ensemble Models (Refer Table 1)	Ensemble Type	Test2022 BLEU
cb-ensemble	Independent	10.32
fb-ensemble	Independent	10.79
ft+ftw+fta+ftaw	Data Augmented Back-translation	10.95
fp+fpw+fpa+fpaw	Data Augmented Paraphrase	**11.26**

Number of models	Avg Test BLEU
4	**10.83**
3	10.60
2	10.23
1 (No Ensemble)	9.24

Table 6: **Impact of Ensembling tmh→fra ST models.** Ensembling models trained with different seeds increases the BLEU score. Increasing the number of models in ensemble also increases performance.

leads to significant performance degradation compared to the ensemble baseline. We attribute this observation to the fact that the pretrained LLMs lacks context-specific data of the Tamasheq corpus. For example, when asked to correct the output sentence, LLMs tend to re-frame the phrases related to more generic topics like sports or events.

Second, we find reranking and token masking strategies both lead to slight degradation compared to the baseline. This is due to the fact that both approaches make less aggressive changes to the original output. In general, we find LLMs do not perform well when the predicted text deviates too much from the ground truth.

Finally, we perform the same set of the strategies but using translated English output from the original French translation. We present the best performing candidates (Translation+Reranking in Table 4). We find that this strategy caused the worst performance degradation due to error propagation

#	Model	Vocab size	Validation BLEU
mwb	wav2vec2-base-960h	1k	11.41
mwbm1k	wav2vec2-base-marathi	1k	**13.19**
mwbm3k	wav2vec2-base-marathi	3k	11.85
mwx	wav2vec2-xls-r-300m	1k	**15.94**
mwxm	wav2vec2-xls-r-300m-marathi	1k	10.76

Table 7: **Model performance on mr→hi task.** Average BLEU scores are shown for the models which we trained with multiple seeds. Move to XLS-R model as encoder improved BLEU by 40% over baseline. Complete results in Table 13

Ensemble Models (Refer Table 7)	Validation BLEU
mwbm1k-ensemble	**16.17**
mwbm3k-ensemble	13.80
mwx-ensemble	**19.63**

Table 8: **Impact of Ensembling mr→hi models.** Consistent with experiments from tmh→fra, an independent ensemble model built from different seeds improves BLEU score.

caused by fra→eng→fra translation.

3.2.4 Marathi-Hindi

We present the BLEU scores of various models we have trained on the validation dataset. From Table 7 we can see that our *wav2vec2-base-marathi* model outperforms the baseline *wav2vec2-base-960h* model by 16% in terms of BLEU score. We also notice increasing vocabulary size of the tokenizer leads to worse performance. It could be attributed to the fact that the size of the data is not adequate for the model to properly train with the provided hyperparameters. The *wav2vec2-xls-r-300m* model outperforms baseline *wav2vec2-base-960h* model by 40%. We notice that the Marathi fine-tuned version of the same model performs worse than our baseline.

We perform independent ensemble decoding on the models with the same architecture and hyperparameters but trained with different seeds. The results are shown in Table 8. Refer Table 14 for full results. We notice that ensemble decoding improves the BLEU score of the best model by 23% compared to the average BLEU score of the individual models used in the ensemble.

3.3 Test 2023 results

Results for the different models on Test 2023 dataset for Tmh→Fra are present in Table 9 and Mr→Hi results are present in Table 10.

Ensemble Models (Refer Table 1)	Ensemble Type	Test2023 BLEU
cb-ensemble	Independent	9.28
fb-ensemble	Independent	**9.50**
ft+ftw+fta+ftaw	Data Augmented Back-translation	8.87
fp+fpw+fpa+fpaw	Data Augmented Paraphrase	9.30

Table 9: **Test 2023 results for tmh→fra ST models.**

Models	Test2023 BLEU
mwbm1k-ensemble	25.60
mwbm3k-ensemble	23.00
mwx-ensemble	**28.60**

Table 10: **Test 2023 results for mr→hi ST models.**

4 Conclusion

In this paper, we explore multiple types of strategies to improve speech translation for two language pairs: Tamasheq-French (Tmh→Fra) and Marathi-Hindi (Mr→Hi). We show expanding the training dataset with paraphrases of translated sentences as well as an ensemble model (of trained ST models with different seeds and data augmentation methods), improves performance over the baseline model for (Tmh→Fra). Similarly, an ensemble model for Marathi-Hindi (Mr→Hi) has a higher BLEU score in comparison to the baseline architecture. We also explore the use of large language models and find that post-processing using them did not show any noticeable improvement.

References

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.

Antonios Anastasopoulos, Loic Barrault, Luisa Bentivogli, Marcely Zanon Boito, Ondrej Bojar, Roldano Cattoni, Anna Currey, Georgiana Dinu, Kevin Duh, Maha Elbayad, Clara Emmanuel, Yannick Esteve, Marcello Federico, Christian Federmann, Souhir Gahbiche, Hongyu Gong, Roman Grundkiewicz, Barry Haddow, Benjamin Hsu, David Javorsky, Vera Kloudova, Surafel Melaku Lakew, Xutai Ma, Prashant Mathur, Paul McNamee, Kenton Murray, Maria Nădejde, Satoshi Nakamura, Matteo Negri, Jan Niehues, Xing Niu, John Ortega, Juan Pino, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stuker, Katsuhito Sudoh, Marco Turchi, Yogesh Virkar, Alex Waibel, Changhan Wang, and Shinji Watanabe. 2022. Findings of the iwslt 2022 evaluation campaign. In *IWSLT 2022*.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations.

Sameer Bansal, Herman Kamper, Karen Livescu, Adam Lopez, and Sharon Goldwater. 2018. Low-resource speech-to-text translation. *arXiv preprint arXiv:1803.09164*.

Joydeep Bhattacharjee. 2022. Xls-r marathi pre-trained model. `https://huggingface.co/infinitejoy/wav2vec2-large-xls-r-300m-marathi-cv8`. Accessed: 2023-04-15.

Nidhir Bhavsar. 2022. French paraphrase model. `https://huggingface.co/enimai/mbart-large-50-paraphrase-finetuned-for-fr`. Accessed: 2023-04-12.

Nidhir Bhavsar, Rishikesh Devanathan, Aakash Bhatnagar, Muskaan Singh, Petr Motlicek, and Tirthankar Ghosal. 2022. Team innovators at SemEval-2022 for task 8: Multi-task training with hyperpartisan and semantic relation for multi-lingual news article similarity. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1163–1170, Seattle, United States. Association for Computational Linguistics.

Marcely Zanon Boito, Fethi Bougares, Florentin Barbier, Souhir Gahbiche, Loïc Barrault, Mickael Rouvier, and Yannick Estéve. 2022a. Speech resources in the tamasheq language. *Language Resources and Evaluation Conference (LREC)*.

Marcely Zanon Boito, John Ortega, Hugo Riguidel, Antoine Laurent, Loïc Barrault, Fethi Bougares, Firas Chaabani, Ha Nguyen, Florentin Barbier, Souhir Gahbiche, et al. 2022b. On-trac consortium systems for the iwslt 2022 dialect and low-resource speech translation tasks. *IWSLT*.

Harveen Singh Chadha, Anirudh Gupta, Priyanshi Shah, Neeraj Chhimwal, Ankur Dhuriya, Rishabh Gaur, and Vivek Raghavan. 2022. Vakyansh: Asr toolkit for low resource indic languages.

Yao-Fei Cheng, Hung-Shin Lee, and Hsin-Min Wang. 2021. Allost: Low-resource speech translation without source transcription. *arXiv preprint arXiv:2105.00171*.

Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N Sainath, Zhijeng Chen, and Rohit Prabhavalkar. 2018. An analysis of incorporating an external language model into a sequence-to-sequence model. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5828. IEEE.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yuchen Liu, Long Zhou, Yining Wang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2018. A comparable study on model averaging, ensembling and reranking in nmt. In *Natural Language Processing and Chinese Computing*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Chenggang Mi, Lei Xie, and Yanning Zhang. 2022. Improving data augmentation for low resource speech-to-text translation with diverse paraphrasing. *Neural Networks*, 148:194–205.

Language Processing LLP Panlingua. 2023. Dataset for marathi-hindi speech translation shared task@iwslt-2023. Contributor/©holder: Panlingua Language Processing LLP, India and Insight Centre for Data Analytics, Data Science Institue, University of Galway, Ireland.

Mihaela C Stoian, Sameer Bansal, and Sharon Goldwater. 2020. Analyzing asr pretraining for low-resource speech-to-text translation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7909–7913. IEEE.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Anjali Chourdia, Artyom Astafurov, Caroline Chen, Ching-Feng Yeh, Christian Puhrsch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z. Yang, Jason Lian, Jay Mahadeokar, Jeff Hwang, Ji Chen, Peter Goldsborough, Prabhat Roy, Sean Narenthiran, Shinji Watanabe, Soumith Chintala, Vincent Quenneville-Bélair, and Yangyang Shi. 2021. Torchaudio: Building blocks for audio and speech processing. *arXiv preprint arXiv:2110.15018*.

Ziqiang Zhang and Junyi Ao. 2022. The YiTrans speech translation system for IWSLT 2022 offline shared task. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 158–168, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

A Appendix

A.1 Hyperparameters and Computing Resource

- encoder
  - n layers: 6
  - hidden dim: 1024 for mr-hi xls-r model, 768 for tmh-fra model and other mr-hi model
  - n head: 12
  - activation: gelu
- decoder
  - n layers: 2
  - hidden dim: 256
  - n head: 4
  - activation: gelu
- training
  - optimizer: AdamW (Loshchilov and Hutter, 2019)
  - lr: $1e-3$
  - encoder lr: $1e-5$
  - label smoothing: 0.1
  - batch size: 4
- computing resource: AWS g5.12xlarge instance (4x NVIDIA A10G Tensor Core GPUs)

A.2 Full Results

#	Data	Data Augmentation	Vocab size	Loss	Seed	Test2022 BLEU
cb1	clean	baseline	1k	baseline	v1	8.98
cb2	clean	baseline	1k	baseline	v2	8.91
cb3	clean	baseline	1k	baseline	v3	8.82
cb4	clean	baseline	1k	baseline	v4	8.69
fb1	full	baseline	1k	baseline	v1	9.53
fb2	full	baseline	1k	baseline	v2	9.10
fb3	full	baseline	1k	baseline	v3	9.21
fb4	full	baseline	1k	baseline	v4	9.17

Table 11: Results of different seed experiments on tmh→fra models.

Data	Data Augmentation	Models in Ensemble (Refer to Table 1)	Test BLEU
clean	baseline	cb1+cb2+cb3+cb4	10.32
clean	baseline	cb1+cb2+cb3	10.22
clean	baseline	cb1+cb2+cb4	9.97
clean	baseline	cb1+cb3+cb4	10.17
clean	baseline	cb2+cb3+cb4	10.14
clean	baseline	cb1+cb2	9.79
clean	baseline	cb1+cb3	9.76
clean	baseline	cb1+cb4	9.86
clean	baseline	cb2+cb3	9.93
clean	baseline	cb2+cb4	10.17
clean	baseline	cb3+cb4	9.67
full	baseline	fb1+fb2+fb3+fb4	10.79
full	baseline	fb1+fb2+fb3	10.52
full	baseline	fb1+fb2+fb4	10.69
full	baseline	fb1+fb3+fb4	10.58
full	baseline	fb2+fb3+fb4	10.42
full	baseline	fb1+fb2	10.00
full	baseline	fb1+fb3	10.16
full	baseline	fb1+fb4	10.22
full	baseline	fb2+fb3	10.08
full	baseline	fb2+fb4	9.98
full	baseline	fb3+fb4	10.06
full	back-translation	ft+ftw+fta+ftaw	10.95
full	back-translation	ft+ftw+fta	10.49
full	back-translation	ft+ftw+ftaw	10.75
full	back-translation	ft+fta+ftaw	10.93
full	back-translation	ftw+fta+ftaw	11.26
full	back-translation	ft+ftw	10.08
full	back-translation	ft+fta	9.82
full	back-translation	ft+ftaw	10.49
full	back-translation	ftw+fta	10.4
full	back-translation	ftw+ftaw	10.72
full	back-translation	fta+ftaw	10.78
full	paraphrase	fp+fpw+fpa+fpaw	11.26
full	paraphrase	fp+fpw+fpa	10.78
full	paraphrase	fp+fpw+fpaw	10.91
full	paraphrase	fp+fpa+fpaw	10.77
full	paraphrase	fpw+fpa+fpaw	11.95
full	paraphrase	fp+fpw	10.40
full	paraphrase	fp+fpa	10.62
full	paraphrase	fp+fpaw	10.76
full	paraphrase	fpw+fpa	10.60
full	paraphrase	fpw+fpaw	10.61
full	paraphrase	fpa+fpaw	10.44

Table 12: Impact of Ensembling tmh→fra models (complete).

#	Model	Vocab size	Seed	Validation BLEU
mwbm1k1	wav2vec2-base-marathi	1k	v1	13.19
mwbm1k2	wav2vec2-base-marathi	1k	v2	13.15
mwbm1k3	wav2vec2-base-marathi	1k	v3	**13.39**
mwbm1k4	wav2vec2-base-marathi	1k	v4	13.01
mwbm3k1	wav2vec2-base-marathi	3k	v1	11.63
mwbm3k2	wav2vec2-base-marathi	3k	v2	11.71
mwbm3k3	wav2vec2-base-marathi	3k	v3	11.80
mwbm3k4	wav2vec2-base-marathi	3k	v4	12.26
mwx1	wav2vec2-xls-r-300m	1k	v1	**16.31**
mwx2	wav2vec2-xls-r-300m	1k	v2	15.35
mwx3	wav2vec2-xls-r-300m	1k	v4	16.09
mwx4	wav2vec2-xls-r-300m	1k	v4	16.00

Table 13: Results of different seed experiments on mr→hi models.

Model	Ensemble Models (Refer Table 13)	Validation BLEU
wav2vec2-base-marathi	mwbm1k1+mwbm1k2+mwbm1k3+mwbm1k4	**16.17**
wav2vec2-base-marathi	mwbm1k1+mwbm1k2+mwbm1k3	16.15
wav2vec2-base-marathi	mwbm1k1+mwbm1k2+mwbm1k4	15.85
wav2vec2-base-marathi	mwbm1k1+mwbm1k3+mwbm1k4	15.89
wav2vec2-base-marathi	mwbm1k2+mwbm1k3+mwbm1k4	15.70
wav2vec2-base-marathi	mwbm1k1+mwbm1k2	15.23
wav2vec2-base-marathi	mwbm1k1+mwbm1k3	15.38
wav2vec2-base-marathi	mwbm1k1+mwbm1k4	14.96
wav2vec2-base-marathi	mwbm1k2+mwbm1k3	15.22
wav2vec2-base-marathi	mwbm1k2+mwbm1k4	14.95
wav2vec2-base-marathi	mwbm1k3+mwbm1k4	15.03
wav2vec2-base-marathi	mwbm3k1+mwbm3k2+mwbm3k3+mwbm3k4	13.80
wav2vec2-xls-r-300m	mwx1+mwx2+mwx3+mwx4	**19.63**
wav2vec2-xls-r-300m	mwx1+mwx2+mwx3	19.27
wav2vec2-xls-r-300m	mwx1+mwx2+mwx4	19.00
wav2vec2-xls-r-300m	mwx1+mwx3+mwx4	19.60
wav2vec2-xls-r-300m	mwx2+mwx3+mwx4	19.20
wav2vec2-xls-r-300m	mwx1+mwx2	17.89
wav2vec2-xls-r-300m	mwx1+mwx3	18.66
wav2vec2-xls-r-300m	mwx1+mwx4	18.35
wav2vec2-xls-r-300m	mwx2+mwx3	18.20
wav2vec2-xls-r-300m	mwx2+mwx4	17.79
wav2vec2-xls-r-300m	mwx3+mwx4	18.59

Table 14: Impact of Ensembling mr→hi models (complete).