A WEAKLY-SUPERVISED STREAMING MULTILINGUAL SPEECH MODEL WITH TRULY ZERO-SHOT CAPABILITY

Jian Xue, Peidong Wang*, Jinyu Li*, Eric Sun*

Microsoft Speech Group, Redmond, WA, USA

ABSTRACT

Streaming automatic speech recognition (ASR) and speech translation (ST) tasks have extensively utilized neural transducers. In this paper, we present our endeavor to construct a Streaming Multilingual Speech Model (SM^2), which employs a single neural transducer model for transcribing or translating multiple languages into target languages. SM^2 is trained using weakly supervised data created by converting speech recognition transcriptions with a machine translation model. Leveraging 351 thousand hours of speech training data from 25 languages, SM^2 achieves impressive ST performance. Furthermore, we demonstrate the truly zero-shot capability of SM^2 when expanding to new target languages, generating high-quality zero-shot ST translation for {source-speech, target-text} pairs that were not seen during training.

Index Terms— automatic speech recognition, speech translation, multilingual, zero-shot, streaming

1. INTRODUCTION

With the advancement of end-to-end (E2E) modeling [1], E2E models have emerged as the dominant model structure in automatic speech recognition (ASR) [2, 3, 4, 5] and speech translation (ST) [6, 7, 8, 9]. This trend has motivated many efforts to develop a unified E2E model for multilingual ASR [10, 11, 12] and multilingual ST [13, 14] tasks. Common E2E techniques employed in ASR include Connectionist Temporal Classification (CTC) [15], Attention-based Encoder-Decoder (AED) [16], and recurrent neural network Transducer (RNN-T) [17, 18, 19]. RNN-T, which eliminates the conditional label independence assumption of CTC and provides a more natural streaming solution than AED, is widely used for ASR tasks in real-world streaming applications. Regarding E2E ST, most previous models have relied on AED architectures due to the attention mechanism's ability to address the word reordering challenge in ST. However, despite the introduction of methods such as Monotonic Chunkwise Attention [20], Monotonic Infinite Lookback Attention [21], and Monotonic Multi-head Attention [22, 23], AED models may not be the most suitable choice for streaming ST. In a recent study [24, 25], the Transformer Transducer (T-T) model, which utilizes a streaming Transformer as the encoder in a neural transducer model, has been shown to be a promising solution for streaming ST, offering high translation quality and low latency. In this work, we also adopt the T-T model as the foundational architecture due to its superior translation quality and low-latency properties.

A recent development in the field of multilingual speech modeling is the Whisper model [26], trained on 680 thousand (K) hours of web data with careful elimination of machine-generated transcriptions. It is an offline Transformer AED model [27] capable of performing various tasks, including ASR, ST, spoken language identification (LID), and voice activity detection. The model exhibits good ASR and ST performance when evaluated on tasks not encountered during training, which has been referred to as its zero-shot capability in [26]. However, such a capacity is typically regarded as model robustness in previous studies [28], and zero-shot translation is typically defined as translation between language pairs whose data were not explicitly encountered during model training [29]. Hence, an ST model with zero-shot translation capability should be trained without exposure to the source-language audio and target-language text pairs.

When building successful speech products in the industry, several additional practical factors must be considered, including streaming capability, inference cost, scalability of language expansion, and scarcity of training data. In line with the pursuit of practical speech product development, we introduce the Streaming Multilingual Speech Model (SM^2), which can transcribe or translate multiple spoken languages into the target language transcription without requiring source language identification. SM^2 differs from [26] in the following key aspects:

1. SM^2 is a streaming model that offers broader applicability and significantly smaller model size, in line with the principles of Green AI [30].

2. By eliminating the need for source LID, SM^2 can recognize and translate code-switched utterances with high quality.

3. The ST training process in SM^2 is entirely weakly supervised, bypassing the reliance on human-labeled parallel corpora.

* Equal Contribution

979-8-3503-0689-7/23/$31.00

4. With only a minimal increase in footprint, SM^2 can be seamlessly expanded to support additional target languages.

5. SM^2 has a truly zero-shot ST capability, enabling it to perform ST without prior training on {source-speech, target-text} pairs.

2. STREAMING MULTILINGUAL SPEECH MODEL

In this section we begin by introducing SM^2 as a model which was originally designed to output texts in a single target language. Following that, we describe the process of expanding SM^2 by incorporating additional output branches, allowing it to generate texts in multiple target languages. Additionally, we will explore how this design facilitates the capability of SM^2 to perform zero-shot ST.

2.1. Streaming Multilingual Speech Model with Single Language Output

Our initial objective in developing SM^2 was to create a single streaming E2E speech model that can transcribe utterances in the target language, such as English, while also is able to translate various spoken languages (excluding English) into the target language (English). In this way, regardless of the language spoken by the user, the system would consistently provide text output in the target language. Our approach differs from the one described in [26], which relies on user input to determine whether to use ASR or ST. Another distinction is that [26] leverages offline processing to identify the language spoken by the user by analyzing the entire utterance. Such LID information is then utilized to guide both ASR and ST processes. The incorporation of LID information plays a crucial role in enhancing the overall quality of speech modeling [11, 31]. Nevertheless, streaming speech model is not able to achieve this functionality due to latency constraints. Additionally, relying on LID information poses challenges in accurately processing code-switched utterances within the system.

The work in [24] demonstrates that a neural transducer is a effective solution for streaming ST tasks, offering both high translation quality and low latency. One notable advantages of utilizing a neural transducer is its innate ability to address the reordering challenge. By dynamically determining read and write operations at each input feature frame, a neural transducer naturally handles the issue of reordering in the context of ST. The neural Transducer has three components: an encoder network, a prediction network, and a joint network. When the encoder network is an RNN or a Transformer, the neural Transducer is called RNN-T or T-T, respectively.

We build SM^2 with T-T which is shown in Fig. 1. The encoder takes speech input $\mathbf{x}_t$ to produce high-level speech representation $\mathbf{h}_t^{enc}$ while the prediction network takes previous non-*blank* output label $\mathbf{y}_{u-1}$ from T-T to generate high-level representation $\mathbf{h}_u^{pre}$. t and u denote the time and label steps, respectively. The joint network is a feedforward network which combines $\mathbf{h}_t^{enc}$ and $\mathbf{h}_u^{pre}$, and finally outputs the probability $P(\mathbf{y}_u \in \mathbf{Y} \cup \emptyset | \mathbf{x}_{1:t}, \mathbf{y}_{1:u-1})$, where $\mathbf{Y}$ is the vocabulary list and $\emptyset$ denotes the *blank* output.

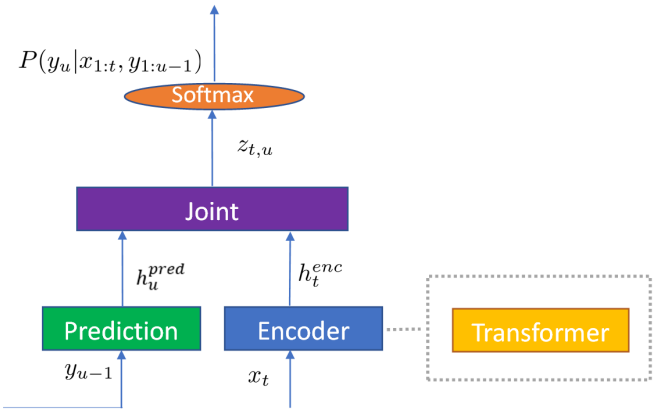

Fig. 1. Illustration of a Transformer-Transducer

We apply the attention mask proposed in [32] for the T-T to work in streaming mode. An example is shown in Fig. 2. We divide the speech inputs into chunks along time with chunk size U. Each frame can see fixed numbers of left chunks, and the left reception field increases linearly with the number of layers, enabling the model to use long history information for a better performance with much less computational cost than the model which uses full history at every layer. Within a chunk, all frame can see each other, but cannot see any frames in future chunks. Therefore, the algorithmic latency of such a T-T is the chunk size U.

In the experiment section, we will employ a chunk size U to regulate the accessibility of future speech frames for the T-T model. Adjusting the chunk size will influence the ASR and ST qualities. A larger chunk size provides the model with more information at each time step, ultimately resulting in improved ASR and ST performance.

During the training of SM^2, we aggregate speech data from various languages into a single pool. When a speech sample belongs to the target language, it is considered as an ASR task, while samples from other languages are treated as ST tasks. SM^2 does not require explicit information regarding whether the task is ASR or ST. Additionally, the system operates without relying on LID information, enabling it to naturally process code-switched utterances with high quality. It is important to acknowledge that acquiring a large-scale human labeled training set for ST is considerably more challenging compared to ASR. To overcome the scarcity of ST training data, we employ a weakly supervised approach [33] by calling a text based machine translation service to translate the ASR transcriptions into the target language. By utilizing

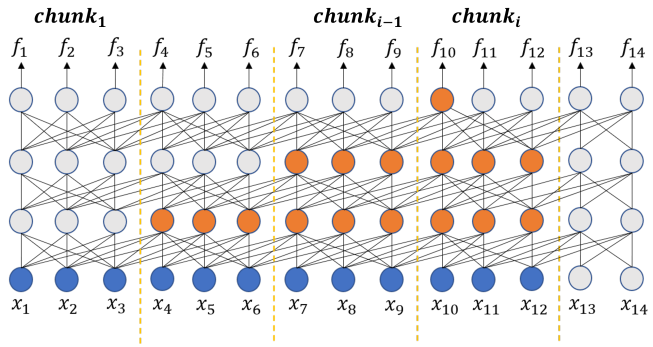

Fig. 2. The reception field of a streaming T-T for generating output f_{10}. The chunk size is 3 and the number of left chunks is 1.

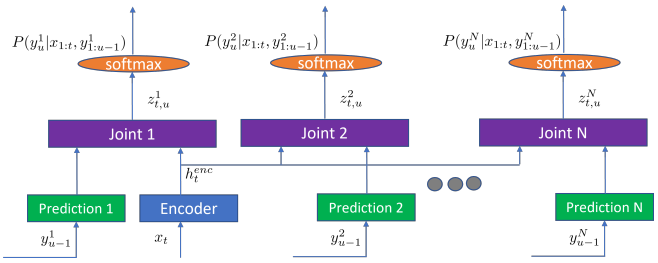

Fig. 3. Illustration of language expansion.

this method, we eliminate the need for any human-labeled ST data during the model training process.

2.2. Language Expansion with Zero-Shot Capability

Scaling to more output languages is challenging to multilingual E2E ST models including SM^2. Suppose we have S source languages and the target language is English, we only need to use S language pairs to train a SM^2. However, if we want to support all S-language outputs, we need to have S^2 language pairs in the training set, introducing formidable training cost. Furthermore, after expanding to more output languages, we would like to avoid degrading the model performance on the original target language.

We propose a language expansion technique as shown in Fig. 3. We first train a SM^2 with one target language using the method described in Section 2.1. When expanding to a new output language, we reuse and freeze the speech encoder from the previous model, and add new prediction and joint networks. Since prediction and joint networks have much less parameters compared with the encoder, the model size increase for adding a new target language is small. The ST training data is again synthesized from the same ASR training corpus as what has been done for the first target language.

Our proposed method enables zero-shot ST, reducing the number of language pairs required during training, and thus drastically improve the training efficiency. Fig. 4 shows the

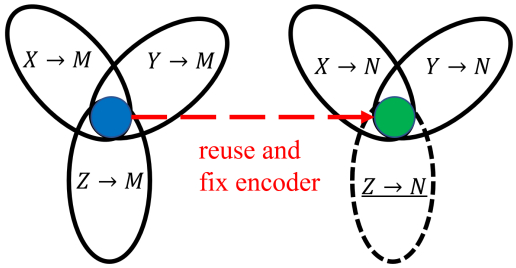

Fig. 4. Illustration of the zero-shot mechanism. $Z \rightarrow N$ is not observed during training.

mechanism facilitating the zero-shot capability of SM^2. For a many-to-one SM^2 trained using $\{X, Y, Z\} \rightarrow M$ data where X, Y, Z, M are different languages, we denote the shared representation space from the speech encoder as a blue circle, in which utterances in different languages have the same semantic meaning. Such inter-lingual space [29] can be obtained when we have a large amount of speech training data in multiple languages. For a new language output N, since we use the same multilingual ASR corpus to generate the transcriptions for M and N and we reuse and freeze the original speech encoder, its inter-lingual space represented by the green circle is the same as that of $\{X, Y, Z\} \rightarrow M$. Therefore, when we train the model for the new target language N only with $\{X, Y\} \rightarrow N$ data, utterances in the inter-lingual space can also perform $Z \rightarrow N$ translation. Because of this calibration in the inter-lingual space and encoder freezing, $Z \rightarrow N$ translation can generalize to other utterances in language Z shown in the dashed area in Fig. 4, thus enables zero-shot translation.

3. EXPERIMENTS

To train SM^2, we use ASR training data from 25 languages: English (EN), Chinese (ZH), Portuguese (PT), Spanish (ES), Italian (IT), German (DE), French (FR), Japanese (JA), Russian (RU), Korean (KO), Polish (PL), Norwegian (NB), Hungarian (HU), Greek (EL), Czech (CS), Romanian (RO), Swedish (SV), Danish (DA), Finnish (FI), Dutch (NL), Slovenian (SL), Slovak (SK), Lithuanian (LT), Estonian (ET), and Bulgarian (BG). As shown in Table 1, the corpora cover lower-, medium-, and high-resource languages containing [0.1K, 1K), [1K, 10K), and [10K, 100K] hours of training data, respectively. The total number of training data is 351K hours. All the training data is anonymized with personally identifiable information removed. A text based machine translation (MT) service is used to convert the ASR transcriptions into texts of the target language for ST training.

We first trained several models based on the T-T structure described in Section 2.1 with same model structure but different algorithmic latencies (encoder lookahead). The above 25 languages are source input language and English is the tar-

hours	languages
[0.1K, 1K)	SL, SK, LT, ET, BG
[1K, 10K)	RU, KO, PL, NB, HU, EL, CS, RO, SV, DA, FI, NL
[10K, 100K]	EN, ZH, PT, ES, IT, DE, FR, JA

Table 1. Training data amount of 25 languages.

get language. The encoder has 36 Transformer blocks, each contains 512 hidden nodes, 8 attention heads, and 4096 feedforward nodes. The prediction network has 2 LSTM layers with 1024 embedding dimension and 1024 hidden nodes. The joint network is a single feedforward layer with 512 nodes and the vocabulary size is 5K. The total number of parameters is 211 million (M). We investigated several chunk sizes as 0.32s, 1s, and 30s. We also trained another larger model with 30s chunk size, which has 24 Transformer blocks, each contains 1024 hidden nodes, 16 attention heads, and 4096 feedforward nodes. The total number of parameters for this model is 343M. The models with 30s chunk size are not feasible in a streaming system. We train such models as comparisons to see the up limit of the accuracy when we keep increasing the latency of the system. The 25 language to ZH model is based on the 211M model with 0.32s latency. The additional number of parameters added specifically for ZH output is 27M.

3.1. Generating English Transcription from 25 Spoken Languages

To compare the ST performance with the model in [26], we take CoVoST 2 [34] as the benchmark and evaluate BLEU scores for both systems. The initial purpose of our SM^2 work is to build an in-house multilingual speech model, therefore we did not select the same language set as in CoVoST 2 and **did not include any CoVoST 2 data in training**. We can only evaluate a subset of 12 language pairs that are observed in our training, as shown in Table 2. The BLEU scores of the MT service which we used to generated the training data are listed in the last column of the Table. The low-latency streaming SM^2 with 211M parameters and 0.32s chunk size has a BLEU score of 28.7 on average, much better than the small model in [26] which has 244M parameters and 30s chunk size[1]. As we keep the model size but increase the chunk size, the SM^2 get better BLEU scores, 31.3 for the one with 1s chunk size, and 32.8 with 30s chunk size. Finally, increasing the number of parameters to 343M and the chunk size to 30s, the SM^2 reaches 33.7 BLEU score, slightly better than the largest model in [26], which has 1550M parameters and 30s chunk size.

Because [26] uses in-house training data, there is no apple-to-apple comparison between these models. However, we observe that

- State-of-the-art ST results can be achieved using weakly supervised ST training data, which is obtained by translating ASR transcriptions to texts of the target language with an MT system, without the need of any human labeled ST data.

- T-T based streaming multilingual ST models can yield very high translation quality even with a small model size and low latency, and without source LID information.

[1]The models in [26] are offline models, but are operated in 30s chunks during inference.

	Whisper		SM^2				MT
model size (M)	244	1550	211			343	NA
chunk size	30s	30s	0.32s	1s	30s	30s	NA
DE→EN	25.3	36.3	32.3	34.0	36.4	**37.8**	45.6
ZH→EN	6.8	18.0	15.9	18.0	19.8	**21.6**	30.5
JA→EN	17.3	**26.1**	20.1	21.6	23.5	25.4	28.4
RU→EN	30.9	43.3	36.8	39.8	43.3	**44.8**	57.4
NL→EN	28.1	41.2	36.1	38.5	42.2	**43.4**	48.5
ET→EN	2.4	15.0	15.3	17.9	21.3	**22.3**	30.7
SV→EN	29.9	**42.9**	33.6	37.1	36.5	33.8	56.2
SL→EN	9.2	21.6	15.3	**22.4**	18.1	20.4	43.9
ES→EN	33.0	**40.1**	32.9	34.7	36.8	37.3	45.8
FR→EN	27.3	**36.4**	31.5	33.0	34.9	35.9	48.0
IT→EN	24.0	30.9	31.7	33.4	35.0	**36.1**	44.5
PT→EN	40.6	**51.6**	42.4	44.7	45.6	45.8	55.0
Average	22.9	33.6	28.7	31.3	32.8	**33.7**	44.5

Table 2. BLEU score comparison of different models on CoVoST 2 tasks with languages→EN observed during training. The **bold** numbers indicate the best BLEU score for a specific language pair.

	SM^2				ASR	
model size	211M			343M	211M	343M
chunk size	0.32s	1s	30s	30s	0.32s	0.32s
WER	8.81	8.18	7.55	7.27	7.72	7.36

Table 3. WERs of SM^2 and ASR models on 1.8M word test sets

We compare different SM^2 variations in Table 3 using our in-house ASR test set, which contains 1.8M words from various tasks. We also trained two ASR models as comparisons, with 0.32s chunk size and different model sizes, which can only transcribe English utterances. The 211M-parameter and 343M-parameter ASR models have the same T-T model structures as the SM^2 variations with the same model size, except that the chunk size may be different. For SM^2, both the 1s and 30s chunk size models are significantly better than the 0.32s model, showing the advantage of larger encoder lookahead. The ASR models with 0.32s chunk size outperform the corresponding SM^2 with the same chunk size in terms of WERs. This indicates that simply merging the transcriptions of ASR and ST together to train a single model is not optimal because the goal of ASR task is to precisely transcribe every word in the spoken utterance, whereas the goal of ST task is to convey the semantic meaning of an utterance.

# source languages	1	3	12	21	25
DE→ZH	**2.2**	21.0	21.8	22.5	21.3
EN→ZH	**0.1**	28.9	29.2	29.3	28.2
JA→ZH	**4.5**	**11.4**	20.0	20.2	20.2
RU→ZH	**8.9**	**20.1**	27.8	28.3	26.8
NL→ZH	**3.5**	**18.4**	**22.6**	24.5	23.9
ET→ZH	**3.9**	**9.7**	**12.4**	14.0	13.1
SV→ZH	**5.8**	**19.3**	**22.4**	23.4	23.1
SL→ZH	**2.1**	**6.3**	**8.1**	8.5	8.7
ES→ZH	**2.0**	**17.3**	**22.3**	**22.8**	25.0
FR→ZH	**2.9**	**16.0**	**20.7**	**21.7**	23.8
IT→ZH	**2.3**	**16.4**	**21.0**	**22.2**	24.2
PT→ZH	**5.1**	**21.6**	**26.4**	**27.0**	28.8
Average	**3.6**	17.2	21.2	22.0	22.3

Table 4. BLEU score comparison among Chinese-output models trained with different numbers of source languages. The **bold** numbers indicate zero-shot evaluations, i.e., the {source-speech, target-text} pairs are not observed during training.

3.2. Language Expansion to Chinese with Zero-Shot Capability

We evaluate the zero-shot capability when expanding the target language to ZH. We defined 5 training sets with different numbers of source languages as shown in Table 4. The models were trained by reusing and freezing the encoder of the 25→EN model which has 211M parameters and 0.32s latency. Then we train a new joint network and a new prediction network for ZH, which has the same structure as the 25→EN model except that the vocabulary size is 15K. The model in the 1-source column was trained with only ZH speech data, and that in the 3-source column used ZH, EN, and DE speech data. For the 12-source column, the model was trained with ZH, EN, DE, CS, EL, HU, NB, PL, RO, RU, JA, and KO. The model in the 21-source column used the speech from all languages except ES, FR, IT, and PT. All these setups have missing {source-speech, target-text} pairs, indicated by the **bold** font in Table 4. The language pairs used for training are selected randomly. We leave the investigation on language selection for zero-shot ST as future work. The model in the 25-source column was trained with the speech from the full 25-language set.

As the number of source languages increases, the average BLEU scores keep improving. When the training data only has ZH speech, the ST quality is low, with an average BLEU score of 3.6. In contrast, with only 3 source languages, SM^2 can already obtain 17.2 average BLEU score, close to the 22.3 score obtained using all 25 languages in training. When a half set of languages are observed during training (the 12-source column), the resulting average BLEU score is 21.2, only 1.1 away from the model trained with the full set of 25 languages. Note that in this 12-source setup, 8 out 12 test language pairs are not observed during training. Going from the 12-source column to the 21-source column and then the 25-source column, we observed that new language pairs for training only give very limited BLEU score boosts from the zero-shot setups, e.g., 22.6 to 24.5 for NL→ZH and 22.8 to 25.0 for ES→ZH. This clearly demonstrates the zero-shot power of our models.

model size	211M			343M
chunk size	0.32s	1s	30s	30s
AP	0.69	0.76	1.0	1.0
AL	1443	1870	5766	5766
DAL	1423	1811	3458	3454

Table 5. Latency comparisons of SM^2 models on Covost2 sets, where AL and DAL values are in milliseconds (ms)

3.3. Latency Measurement

To assess the inference latencies of our SM^2 models, we utilize three metrics: average proportion (AP), average lagging (AL), and differentiable average lagging (DAL), as proposed in [35]. Table 5 provides an overview of the latency results, with all numbers representing averages across the CoVost2 sets mentioned in Table 2. It is worth noting that the models with a chunk size of 30 seconds effectively function as offline models, given that the average audio length in the test set is approximately 5.7 seconds.

4. CONCLUSIONS

This paper introduces our development of the **S**treaming **M**ultilingual **S**peech **M**odel (SM^2), a unified model that handles both ASR and ST tasks without necessitating explicit task specifications from users. To achieve streaming capability, We adopted the Transformer Transducer as the underlying model architecture and regulated the model latency by adjusting the chunk size in the speech encoder. Notably, no human-labeled ST data was employed during training. It was purely weakly supervised ST data generated by converting 351K hours of anonymized ASR data from 25 languages using text based machine translation service. We designed a language expansion strategy that introduces a minimal number of parameters to the original model. This strategy empowers the model with true zero-shot capability, allowing it to handle previously unseen {source-speech, target-text} pairs by leveraging interlingua representations. The incorporation of these representations enables effective translation between languages without requiring prior training on specific language pairs.

For the task of generating English translations, the SM^2 with 0.32s algorithmic latency obtained much better BLEU score as the model with similar size (211M parameters vs. 244M parameters) in [26], which is not streaming. The best

SM^2 got similar BLEU score as the largest model in [26], but model size is less than $1/4$ of that model. Finally, we demonstrated the strong zero-shot capability of SM^2 when expanding to support the Chinese output. The model trained with only half of language pairs is only 1.1 BLEU score behind the model trained with the full language pairs.

Based on our experiments, we observed that directly merging ASR and ST texts to train a single model may not yield optimal results due to the distinct objectives of ASR and ST. In our future endeavors, we aim to investigate more effective training methods that can tackle this challenge and drive further advancements in SM^2.

5. REFERENCES

[1] J. Li, "Recent advances in end-to-end automatic speech recognition," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.

[2] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[3] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, et al., "State-of-the-art speech recognition with sequence-to-sequence models," in *Proceedings of ICASSP*, 2018, pp. 4774–4778.

[4] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, et al., "Streaming end-to-end speech recognition for mobile devices," in *Proceedings of ICASSP*, 2019, pp. 6381–6385.

[5] J. Li, Y. Wu, Y. Gaur, C. Wang, R. Zhao, and S. Liu, "On the comparison of popular end-to-end models for large scale speech recognition," in *Proceedings of Interspeech*, 2020, pp. 1–5.

[6] A. Berard, O. Pietquin, C. Servan, and L. Besacier, "Listen and translate: A proof of concept for end-to-end speech-to-text translation," in *NIPS Workshop on End-to-end Learning for Speech and Audio Processing*, 2016.

[7] L. C. Vila, C. Escolano, J. A. Fonollosa, and M. R. Costa-Jussa, "End-to-end speech translation with the transformer," in *Proceedings of Interspeech*, 2018, pp. 60–63.

[8] J. Pino, L. Puzon, J. Gu, X. Ma, A. D. McCarthy, and D. Gopinath, "Harnessing indirect training data for end-to-end automatic speech translation: Tricks of the trade," *arXiv preprint arXiv:1909.06515*, 2019.

[9] M. Sperber and M. Paulik, "Speech translation and the end-to-end promise: Taking stock of where we are," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7409–7421.

[10] S. Watanabe, T. Hori, and J. R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *Proceedings of ASRU*, 2017, pp. 265–271.

[11] S. Toshniwal, T. N. Sainath, R. J. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, "Multilingual speech recognition with a single end-to-end model," in *Proceedings of ICASSP*, 2018, pp. 4904–4908.

[12] L. Zhou, J. Li, E. Sun, and S. Liu, "A configurable multilingual model is all you need to recognize all languages," in *Proceedings of ICASSP*, 2022, pp. 6422–6426.

[13] H. Inaguma, K. Duh, T. Kawahara, and S. Watanabe, "Multilingual end-to-end speech translation," in *Proceedings of ASRU*, 2019, pp. 570–577.

[14] X. Li, C. Wang, Y. Tang, C. Tran, Y. Tang, J. Pino, A. Baevski, A. Conneau, and M. Auli, "Multilingual speech translation with efficient finetuning of pretrained models," *arXiv preprint arXiv:2010.12829*, 2020.

[15] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[16] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[17] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[18] R. Prabhavalkar, K. Rao, T. N. Sainath, B. Li, L. Johnson, and N. Jaitly, "A comparison of sequence-to-sequence models for speech recognition," in *Proceedings of Interspeech*, 2017, pp. 939–943.

[19] G. Saon, Z. Tüske, D. Bolanos, and B. Kingsbury, "Advancing rnn transducer techology for speech recognition," in *Proceedings of ICASSP*, 2021, pp. 5654–5658.

[20] C.-C. Chiu and C. Raffel, "Monotonic chunkwise attention," in *International Conference on Learning Representations*, 2018.

[21] N. Arivazhagan, C. Cherry, W. Macherey, C.-C. Chiu, S. Yavuz, R. Pang, W. Li, and C. Raffel, "Monotonoic infinite lookback attention for simultaneous machine translation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1313–1323.

[22] X. Ma, J. Pino, J. Cross, L. Puzon, and J. Gu, "Monotonic multihead attention," in *Proceedings of International COnference on Learning Representations*, 2019.

[23] X. Ma, Y. Wang, M. J. Dousti, P. Koehn, and J. Pino, "Streaming simultaneous speech translation with augmented memory transformer," in *Proceedings of ICASSP*, 2021, pp. 7523–7527.

[24] J. Xue, P. Wang, J. Li, M. Post, and Y. Gaur, "Large-scale streaming end-to-end speech translation with neural transducers," in *Proceedings of Interspeech*, 2022, pp. 3263–3267.

[25] P. Wang, E. Sun, J. Xue, Y. Wu, L. Zhou, Y. Gaur, S. Liu, and J. Li, "LAMASSU: A streaming language-agnostic multilingual speech recognition and translation model using neural transducers," in *Proc. Interspeech*, 2023.

[26] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 6000–6010.

[28] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 745–777, 2014.

[29] M. Johnson, M. Schuster, Q. V. Le, et al., "Google's multilingual neural machine translation system: Enabling zero-shot translation," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 339–351, 2017.

[30] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green AI," *Communications of the ACM*, vol. 63, no. 12, pp. 54–63, 2020.

[31] A. Kannan, A. Datta, T. N. Sainath, E. Weinstein, B. Ramabhadran, Y. Wu, A. Bapna, Z. Chen, and S. Lee, "Large-scale multilingual speech recognition with a streaming end-to-end model," in *Proceedings of Interspeech*, 2019, pp. 2130–2134.

[32] X. Chen, Y. Wu, Z. Wang, S. Liu, and J. Li, "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," in *Proceedings of ICASSP*, 2021, pp. 5904–5908.

[33] Y. Jia, M. Johnson, W. Macherey, R. J. Weiss, Y. Cao, C.-C. Chiu, N. Ari, S. Laurenzo, and Y. Wu, "Leveraging weakly supervised data to improve end-to-end speech-to-text translation," in *Proceedings of ICASSP*, 2019, pp. 7180–7184.

[34] C. Wang, A. Wu, J. Gu, and J. Pino, "CoVoST 2 and massively multilingual speech translation," in *Proceedings of Interspeech*, 2021, pp. 2247–2251.