INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Self-Training for End-to-End Speech Translation

Juan Pino[1], *Qiantong Xu*[1], *Xutai Ma*[1,2], *Mohammad Javad Dousti*[1], *Yun Tang*[1]

[1]Facebook AI, USA
[2]Johns Hopkins University, USA

{juancarabina,qiantong,dousti,yuntang}@fb.com, xutai_ma@jhu.edu

Abstract

One of the main challenges for end-to-end speech translation is data scarcity. We leverage pseudo-labels generated from unlabeled audio by a cascade and an end-to-end speech translation model. This provides 8.3 and 5.7 BLEU gains over a strong semi-supervised baseline on the MuST-C English-French and English-German datasets, reaching state-of-the art performance. The effect of the quality of the pseudo-labels is investigated. Our approach is shown to be more effective than simply pre-training the encoder on the speech recognition task. Finally, we demonstrate the effectiveness of self-training by directly generating pseudo-labels with an end-to-end model instead of a cascade model.

Index Terms: end-to-end speech translation, self-training.

1. Introduction

Speech translation (ST) systems convert input audio in a language into text translations in another language. Compared with their cascade counterpart, end-to-end models have lower inference latency, are smaller and are less susceptible to error compounding. However, their main disadvantage comes from the lack of supervised training data.

Data scarcity has been addressed in previous work with data augmentation [1, 2], multi-task training [3, 4], pre-training [5, 6] or multilingual speech translation [7, 8, 9]. In this paper, we propose to revisit self-training [10] in the context of speech translation. Labels are automatically generated from unlabeled audio data either via a strong speech recognition (ASR) system followed by a strong machine translation (MT) system, i.e. a cascade model, or via an end-to-end model. An end-to-end speech translation model is then trained on the resulting data.

[11] demonstrates the effectiveness of self-training for machine translation and summarization. They also provide insights into its success and further improve vanilla self-training by introducing noise in the unlabeled data. [12] and [13] also leverage pseudo-labeling on the LIBRILIGHT dataset [14] to improve the performance of an end-to-end ASR system. Similar to this work, additional knowledge (i.e., additional monolingual data to train the language model) is leveraged to generate the pseudo-labels. [15] also explores pseudo-labeling (both knowledge distillation and self-training) at scale in the domain of computer vision. [1] demonstrates how to improve the performance of an end-to-end speech translation system by generating pseudo-labels from unlabeled audio via a cascade system. In contrast, we provide more insights on the conditions under which this method works. Furthermore, we demonstrate how to generate pseudo-labels with an end-to-end system, which may simplify model building. Finally, we conduct experimentation on open benchmarks for reproducibility. [16] also leverages additional ASR and MT resources to improve end-to-end speech translation with a meta-learning algorithm. Our approach aims at simplifying model building by reusing either off-the-shelf ASR and MT systems or an end-to-end speech translation for pseudo-labeling and obtains state-of-the-art results.

Table 1: *Open and FB Video dataset statistics, reported after filtering for too long or too short input.*

Domain	Language	Dataset	# utterances	# hours
Open	En-Fr	MuST-C	275k	479
		dev	1412	2.6
		tst-COMMON	2632	4.2
	En-De	MuST-C	230k	395
		dev	1423	2.5
		tst-COMMON	2641	4.1
	En	LIBRISPEECH	281k	960
		LIBRILIGHT	15.8M	56k
FBVideos	En-Fr	train	20.7	30k
		dev	925	6.3
		test	3909	24.3
	En-Es	train	20.6M	30k
		dev	935	6.4
		test	3915	24.3
	En	unlabeled	32.2M	255k

Our method is first shown to be effective in a low resource setting (§3.1). On a higher resource setup, improvements are obtained after fine-tuning on the baseline data and by training larger models (§3.2). Since pseudo-labeling enables the training of larger architectures, scaling up the size of ST models is investigated next (§3.3). By doing so, we obtain large improvements over a strong semi-supervised baseline across three language pairs and two domains and reach state-of-the-art performance on the MuST-C English-French and English-German datasets. In ablation studies, our method is shown to be more effective than pre-training the encoder on the ASR task (§4.1). We also study the effect of the quality of the pseudo-labels on the low resource setting (§4.2). Finally, replacing the cascade model with an end-to-end model for pseudo-labeling is investigated (§4.3).

2. Experimental Setup

2.1. Data

Experiments are conducted with both open and proprietary data. Open data is used for reproducibility purposes and to conduct more detailed ablations while proprietary data, comprised of de-identified and aggregated public Facebook (FB) videos, is used to verify that our methods work at large scale. Open data includes the English-German and English-French portions of MuST-C [17], LIBRISPEECH [18] (LS) transcripts with automatic translations for a higher resource baseline and LIBRILIGHT (LL) to provide English unlabeled audio. Differ-

http://dx.doi.org/10.21437/Interspeech.2020-2938

ent amounts of English unlabeled data are randomly sampled and reused for all experiments. Three language pairs, English-German (En-De), English-French (En-Fr) and English-Spanish (En-Es) are studied. Dataset statistics are summarized in Table 1.

2.2. Speech Translation Models

Models take log-mel filterbank features, computed with a 10 ms window shift, as input. On FB Video data, features have 40 dimensions and a window size of 16 ms, and utterances of more than 6000 frames are removed. On open data, features have 80 dimensions and a window size of 25 ms, and utterances with more than 4000 frames, less than 20 frames, or more than 256 tokens are removed. The translated text vocabulary is a unigram model with size 10,000 built with the SentencePiece [19]. Note that a separate vocabulary is rebuilt for each data condition and that the model is directly built on raw data without pre-tokenization.

We investigate our proposed method with a relatively small LSTM architecture and a large Transformer architecture [20]. The LSTM architecture consists of a speech encoder with non-linear layers followed by convolutional layers and bidirectional LSTM layers, and a custom LSTM decoder [2, 4]. The Transformer architecture, vggtransformer, is an adaptation of Transformer to the ASR task [21]. Two architectures, VggT, with 14 encoder layers and 4 decoder layers, and VggTLarge, with 20 encoder layers and 10 decoder layers are used in experiments.

Training uses the Adam optimizer [22] with a learning rate of 0.001 for the LSTM architecture and 0.0001 for the vggtransformer architecture. The LSTM architecture has a fixed learning rate schedule while the vggtransformer architecture uses the original Transformer learning rate schedule [20]. Mini-batches have an effective size of 384,000 frames. Models are trained until convergence or up to 800k updates.

At inference time, we use beam search with beam size 20, including for pseudo-label generation. Case-sensitive detokenized BLEU is computed with SacreBLEU [23].

2.3. Speech Recognition Models

All the models take 80-channel log-mel filterbank features as input and are trained end-to-end with the Connectionist Temporal Classification (CTC) criterion [24]. The target vocabulary is a wordpiece model [25] with size 10,000. Models are all trained in the *wav2letter++* framework [26] using either the LibriSpeech dataset or FB Videos.

Model trained on full LS: We use the Transformer model from [12] that works best on the LibriSpeech dataset[1]. Specifically, there are 6 layers of 1-D convolutions with kernel width 3 as front-end followed by 24 4-head Transformer blocks with self-attention dimension 1024. The 2nd, 4th and the last convolutions in the front-end have stride 2, so the overall sub-sampling of the model is 8.

Model trained on LS 100h: We use a similar model as for full LS. In order to obtain better performance with a small amount of training data, we use 24 4-head Transformer blocks with self-attention dimension 768 in the middle.

Model trained on FB Videos: The model is mainly built upon Time-Depth Separable Convolution (TDS) [27] blocks. It is composed of one 2-D convolution layer and two fully-connected layers with ReLU, LayerNorm and residual connections in between. Specifically, the model has 4 groups of TDS blocks with a 1-D convolutions at the beginning of each group as transitions. Similarly, the first 3 convolutions have stride 2 so as to reach the same sub-sampling rate of 8. There are 2, 2, 5, and 8 TDS blocks in each group, containing 16, 16, 24, and 32 channels, respectively. Following [12], we also apply a channel increasing factor $F = 2$ in each TDS block.

Table 2: *Number of parameters for each model architecture.*

Task	Model	# Parameters
ST	LSTM	13.5M
	VggT	260.0M
	VggTLarge	435.0M
ASR	Transformer 1024	339.9M
	Transformer 768	204.7M
	TDS	292.0M
MT	En-Es FB Video	320.1M
	En-Fr FB Video	300.6M
	En-De [29]	209.9M
	En-Fr [29]	221.9M

Language model: A language model (LM) is integrated in the beam-search decoder to generate final transcriptions together with the acoustic models. In our experiments, we use 4-gram LMs trained with KenLM toolkit [28]. The LM used for the FB Videos is trained on the transcriptions, while the one for the LibriSpeech dataset is trained on its official LM corpus excluding books containing the transcriptions of LibriLight dataset audios. The latter LM is prepared in [12].

After the acoustic models converged on the labeled data, we tune the beam-search decoder parameters on the validation set. Specifically, the decoder consumes the posterior from the acoustic model, and runs a beam search through with LM to generate the best path, with beam size 300 for the small Transformer (768) and 500 for the large Transformer (1024).

2.4. Machine Translation Models

All MT models use a Transformer [20] architecture. The model for FB Videos uses 6 encoder layers, encoder embedding dimension of 1024, encoder feed-forward network (FFN) dimension of 2048, 16 attention heads, 2 decoder layers, decoder embedding dimension of 512, decoder FFN dimension of 1024, dropout of 0.2, and label smoothing of 0.1. A bottleneck linear layer with dimension 128 is inserted prior to the softmax over the target vocabulary and the decoder is an average attention network [30]. The final model is an ensemble of size 3 obtained from 3 training runs started with different random seeds, where for each training runs, the last 10 checkpoints are averaged. The original Transformer optimizer settings are used, with an initial learning rate of 0.0007 and an effective batch size of 64,000 tokens. The models are trained on 100M sentence pairs from the web, news and social media domain until convergence. In inference, hypotheses are generated with beam search with beam size 2. LibriSpeech transcripts and LibriLight automatic transcripts are translated with pre-trained[2] English-French and English-German models [29], with beam size 5. Model sizes are summarized in Table 2.

[1] github.com/facebookresearch/wav2letter/tree/master/recipes/models/sota/2019.

[2] github.com/pytorch/fairseq/tree/master/examples/translation

3. Results

In this section, we study under which conditions pseudo-labels generated by the cascade model can benefit the ST model.

3.1. Adding LIBRILIGHT pseudo-labels

We first study the effect of simply adding pseudo-labels to the baseline training data and retraining with the LSTM model, where the baseline is either lower resource (MuST-C) or higher resource (MuST-C + LS). In Figure 1, with this simple method, the low resource baseline can be improved by up to 2.4 BLEU on the En-De MuST-C dev set. Above a certain amount, adding unlabeled data degrades performance. However, the higher resource baseline is not improved by simply adding unlabeled data. Next, our focus is on improving the higher resource baseline and on leveraging larger amounts of unlabeled data.

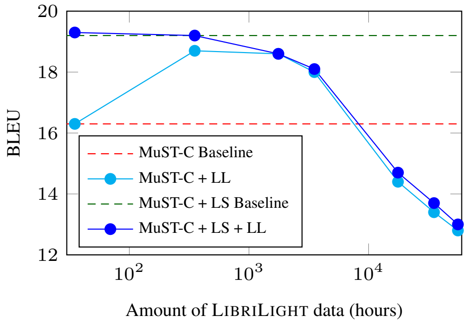

Figure 1: *Results obtained with additional* LIBRILIGHT *(LL) data and the LSTM architecture on the En-De MuST-C dev set.*

3.2. Improving a High-Resource Baseline

In Figure 2, the higher resource baseline can be improved upon by fine-tuning the LSTM model on the baseline data (MuST-C + LS) and by training (and fine-tuning) VGGT. Fine-tuning simply consists in loading the latest checkpoint from the initial training phase and continuing training on the baseline training data (without resetting the optimizer parameters). We obtain up to 24.7 BLEU, i.e., a 5.5 BLEU gain over the high resource baseline. Note that training the VGGT model on the baseline data does not converge and yields only 3.7 BLEU.

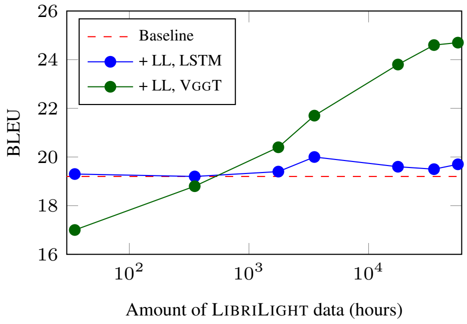

Figure 2: *Improving the higher resource MuST-C + LS baseline on the En-De MuST-C dev set by fine-tuning the LSTM model and training and fine-tuning a larger architecture,* VGGT.

Table 3: *Results obtained on the En-De MuST-C dev set when increasing the model size. Results are reported after fine-tuning.*

Data	VGGT	VggTLarge
MuST-C + LS + 17,607h LL	23.8	23.7
MuST-C + LS + 35,217h LL	24.6	25.6

3.3. Scaling Model Size

In §3.2, substantial improvements over the baseline were obtained by training a larger model. In Table 3, the capacity of the model is further increased in order to verify to what extent pseudo-labels can benefit training. When adding 17,607 h of unlabeled data, VGGT and VGGTLARGE obtain similar performance but with 35,217 h of additional data, VGGTLARGE obtains 1 BLEU improvement on the En-De MuST-C dev set.

3.4. Main Results

We now validate our findings on three languages and two domains. In the En-Fr and En-Es FB Video setting, different amounts of unlabeled data are added to the baseline data, then VGGT is retrained and fine-tuned. Figure 3 confirms earlier conclusions that fine-tuning is necessary to obtain improvements over a strong high-resource baseline. We obtain up to 1.2 and 1.0 BLEU gains on the En-Fr and En-Es dev sets, respectively. Final results on the test sets on three language pairs and two domains are summarized in Table 4. On the En-Fr MuST-C tst-COMMON dataset, we obtain 8.3 BLEU improvements over the strong MuST-C + LS baseline and improve state-of-the-art from [16] by 0.45 BLEU. On the En-De MusT-C tst-COMMON dataset, we obtain 5.7 BLEU improvements over the MusT-C + LS baseline and improve the state-of-the-art by 3.1 BLEU. Finally, we verify that our method works with a very large-scale FB video baseline by obtaining 1.3 and 1.4 BLEU gains on the FB Video En-Fr and En-Es test sets, respectively.

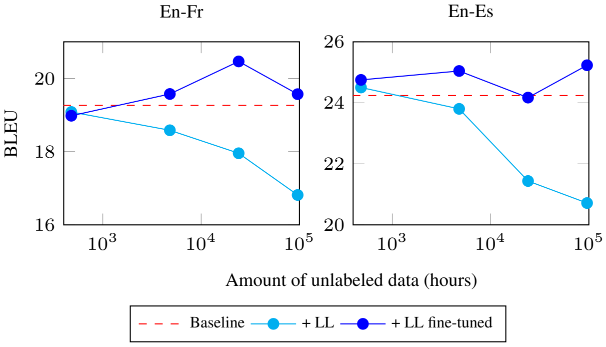

Figure 3: *Effectiveness of pseudo-labeling on FB Videos for En-Fr and En-Es. Results are reported on the dev set.*

4. Ablation Studies

4.1. Pseudo-Labeling vs. ASR Encoder Pre-training

In §3.2, pseudo-labels enable training much larger architectures that are otherwise difficult to train. In this section, we investigate whether this regularization effect is simply due to better pre-training of the encoder. To verify this, the VGGT architecture is first trained on the ASR task exactly as in the

Table 4: *Leveraging unlabeled audio on 3 language pairs and 2 domains. Results are reported on the MuST-C tst-COMMON sets and the FB Video test sets.*

Language	Data	Model	BLEU
En-Fr	MuST-C	LSTM	24.8
	MuST-C + LS		26.2
	MuST-C + LS	VGGT	23.9
	+ 35,217h LL + fine-tuning		**34.5**
	State-of-the-art baseline [16]		34.05
En-De	MuST-C	LSTM	15.6
	MuST-C + LS		19.5
	MuST-C + LS	VGGT	3.5
	+ 35,217h LL + fine-tuning		24.8
	+ 35,217h LL + fine-tuning	VGGTLarge	**25.2**
	State-of-the-art baseline [16]		22.11
En-Fr (FB Videos)	baseline	VGGT	20.3
	+ 96k h unlabeled + fine-tuning		**21.6**
En-Es (FB Videos)	baseline	VGGT	18.5
	+ 96k h unlabeled + fine-tuning		**19.9**

Table 5: *Comparing self-training and pre-training the encoder on the ASR task. Results are reported on the En-De MuST-C dev set after fine-tuning.*

Data	Encoder Pre-training BLEU (WER)	Pseudo-Labeling BLEU
MuST-C + LS	19.8 (25.2)	3.7
+ 3523 h LL	20.8 (22.7)	21.7
+ 17,607 h LL	21.0 (40.3)	23.9

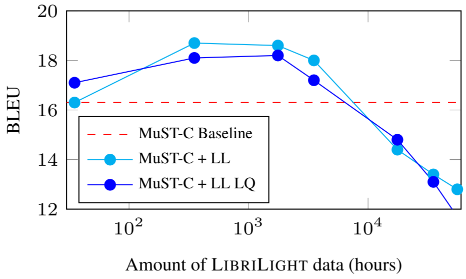

Figure 4: *Effect of using lower quality (LQ) pseudo-labels. Results are reported on the En-De MuST-C dev set.*

Table 6: *Pseudo-labeling with end-to-end speech translation. Results are reported on the En-De MuST-C dev set, after fine-tuning.*

Data	Pseudo-Labeling Model	Model	BLEU
MuST-C	N/A	LSTM	16.3
+ 3523h LL	Cascade	LSTM	20.8
	LSTM		18.5
	VGGT		20.6
MuST-C + LS	N/A	LSTM	19.2
+ 17,607h LL	Cascade	VGGT	23.8
	LSTM		20.7
	VGGT		**24.5**

ST task, then the encoder is initialize with the parameters obtained in ASR training and the same model is trained on the ST task. Data conditions include the high-resource baseline and the baseline augmented with 3523 hours and 17,607 hours from LibriLight. Results are reported in Table 5. The word error rate (WER) obtained on the ASR task is also reported for the encoder pre-training method (the higher WER obtained with 17,607h of data is simply due to the large amount of weakly supervised data but this setting still benefits the ST task). Except in the baseline setting, pseudo-labeling outperforms encoder pre-training, by up to 2.9 BLEU. This highlights the importance of both encoder and decoder pre-training.

4.2. Quality of Pseudo-Labels

The effect of the quality of pseudo-labels is now investigated. Automatic transcripts are generated either with the ASR model trained on the full LibriSpeech dataset or on a 100 hour subset, then translated with the same translation system. The two models obtain 7.3 and 27.7 WER on the LibriSpeech `dev-other` set. The LSTM speech translation model is then retrained on both types of labels with different data amounts. As expected, Figure 4 shows that in the majority of data conditions, the BLEU score increases with higher quality labels.

4.3. Self-Training

So far, pseudo-labels have been generated via a cascade model. In this section, two end-to-end models are considered for pseudo-label generation. The pure self-training scenario where the LSTM end-to-end model has only been trained on the supervised training data is first considered. Pseudo-labels are also generated with the VGGT model trained on MuST-C, LS and 17,607h of cascade-generated LibriLight pseudo-labels. We contrast pseudo-label generation by the cascade model and these two models in Table 6. First, all pseudo-labeling methods improve upon the baseline, even the pure self-training method. The weakest pseudo-labeling method is the pure self-training method that does not use extra information in the process. In the lower resource baseline setting, the cascade and VGGT obtain equivalent performance, the cascade having a slight advantage of 0.2 BLEU. In the higher resource setting, the VGGT end-to-end pseudo-labeling method outperforms the cascade pseudo-labeling by 0.7 BLEU. We conclude that cascade pseudo-labeling can to bootstrap the pseudo-labeling process, then end-to-end ST can be relied on for pseudo-labeling in subsequent iterations.

5. Conclusion

We have shown the effectiveness of pseudo-labels for end-to-end ST in low- and high-resource data conditions, across two domains and 3 language pairs. In the high-resource setting, fine-tuning and larger architectures were found to be critical for obtaining improvements over the baseline. Larger amounts of pseudo-labels allow to increase the model size further. By doing so, we obtained state-of-the-art results on the MuST-C English-French and English-German datasets. Our approach was shown empirically to be more effective than encoder pre-training, highlighting the importance of pre-training the decoder. Finally, pseudo-labeling may be further simplified by utilizing end-to-end ST systems instead of a cascade system.

6. References

[1] 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 *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7180–7184.

[2] 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," in *Proceedings of the 16th International Workshop on Spoken Language Translation (IWSLT)*, 2019.

[3] R. J. Weiss, J. Chorowski, N. Jaitly, Y. Wu, and Z. Chen, "Sequence-to-sequence models can directly translate foreign speech," in *Proc. Interspeech 2017*, 2017, pp. 2625–2629. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2017-503

[4] A. Bérard, L. Besacier, A. C. Kocabiyikoglu, and O. Pietquin, "End-to-end automatic speech translation of audiobooks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6224–6228.

[5] S. Bansal, H. Kamper, K. Livescu, A. Lopez, and S. Goldwater, "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)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 58–68. [Online]. Available: https://www.aclweb.org/anthology/N19-1006

[6] M. C. Stoian, S. Bansal, and S. Goldwater, "Analyzing ASR pretraining for low-resource speech-to-text translation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7909–7913.

[7] M. A. Di Gangi, M. Negri, and M. Turchi, "One-to-many multilingual end-to-end speech translation," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 585–592.

[8] H. Inaguma, K. Duh, T. Kawahara, and S. Watanabe, "Multilingual end-to-end speech translation," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 570–577.

[9] C. Wang, J. Pino, A. Wu, and J. Gu, "Covost: A diverse multilingual speech-to-text translation corpus," *arXiv preprint arXiv:2002.01320*, 2020.

[10] H. Scudder, "Probability of error of some adaptive pattern-recognition machines," *IEEE Transactions on Information Theory*, vol. 11, no. 3, pp. 363–371, 1965.

[11] J. He, J. Gu, J. Shen, and M. Ranzato, "Revisiting self-training for neural sequence generation," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=SJgdnAVKDH

[12] G. Synnaeve, Q. Xu, J. Kahn, T. Likhomanenko, E. Grave, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, "End-to-end ASR: from supervised to semi-supervised learning with modern architectures," in *Proceedings of the ICML 2020 Workshop on Self-supervision in Audio and Speech*, 2020.

[13] J. Kahn, A. Lee, and A. Hannun, "Self-training for end-to-end speech recognition," *arXiv preprint arXiv:1909.09116*, 2019.

[14] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, "Libri-light: A benchmark for ASR with limited or no supervision," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7669–7673.

[15] I. Z. Yalniz, H. Jégou, K. Chen, M. Paluri, and D. Mahajan, "Billion-scale semi-supervised learning for image classification," *arXiv preprint arXiv:1905.00546*, 2019.

[16] S. Indurthi, H. Han, N. K. Lakumarapu, B. Lee, I. Chung, S. Kim, and C. Kim, "End-end speech-to-text translation with modality agnostic meta-learning," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7904–7908.

[17] M. A. Di Gangi, R. Cattoni, L. Bentivogli, M. Negri, and M. Turchi, "MuST-C: a Multilingual Speech Translation Corpus," 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 2012–2017. [Online]. Available: https://www.aclweb.org/anthology/N19-1202

[18] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[19] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *EMNLP*, 2018.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[21] A. Mohamed, D. Okhonko, and L. Zettlemoyer, "Transformers with convolutional context for ASR," *arXiv preprint arXiv:1904.11660*, 2019.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, vol. abs/1412.6980, 2015.

[23] M. Post, "A call for clarity in reporting BLEU scores," in *Proceedings of the Third Conference on Machine Translation: Research Papers*. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 186–191. [Online]. Available: https://www.aclweb.org/anthology/W18-6319

[24] 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.

[25] M. Schuster and K. Nakajima, "Japanese and Korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5149–5152.

[26] V. Pratap, A. Hannun, Q. Xu *et al.*, "wav2letter++: The fastest open-source speech recognition system," *arXiv preprint arXiv:1812.07625*, 2018.

[27] A. Hannun, A. Lee, Q. Xu, and R. Collobert, "Sequence-to-sequence speech recognition with time-depth separable convolutions," *Interspeech 2019*, Sep 2019.

[28] K. Heafield, "KenLM: Faster and smaller language model queries," in *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Edinburgh, Scotland: Association for Computational Linguistics, Jul. 2011, pp. 187–197. [Online]. Available: https://www.aclweb.org/anthology/W11-2123

[29] M. Ott, S. Edunov, D. Grangier, and M. Auli, "Scaling neural machine translation," in *Proceedings of the Third Conference on Machine Translation: Research Papers*. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 1–9. [Online]. Available: https://www.aclweb.org/anthology/W18-6301

[30] B. Zhang, D. Xiong, and J. Su, "Accelerating neural transformer via an average attention network," *arXiv preprint arXiv:1805.00631*, 2018.