End-to-End Offline Speech Translation System for IWSLT 2020 using Modality Agnostic Meta-Learning

Nikhil Kumar Lakumarapu*, Beomseok Lee*, Sathish Indurthi,
Houjeung Han, Mohd Abbas Zaidi, Sangha Kim
Next AI Solution Lab, Samsung Research, Seoul, South Korea
{n07.kumar, bsgunn.lee, s.indurthi, h.j.han, abbas.zaidi, sangha01.kim}@samsung.com

Abstract

In this paper, we describe the system submitted to the IWSLT 2020 Offline Speech Translation Task. We adopt the Transformer architecture coupled with the meta-learning approach to build our end-to-end Speech-to-Text Translation (ST) system. Our meta-learning approach tackles the data scarcity of the ST task by leveraging the data available from Automatic Speech Recognition (ASR) and Machine Translation (MT) tasks. The meta-learning approach combined with synthetic data augmentation techniques improves the model performance significantly and achieves BLEU scores of 24.58, 27.51, and 27.61 on IWSLT test 2015, MuST-C test, and Europarl-ST test sets respectively.

1 Introduction

The goal of the IWSLT 2020 Offline Speech Translation challenge(Ansari et al., 2020) is to check the feasibility of end-to-end models for translating audio speech of one language into text of a different target language. The success of end-to-end neural models for ASR (Graves et al., 2013) and MT (Bahdanau et al., 2015) inspired to build end-to-end neural models for the more challenging Speech-to-Text translation (ST) task (Bérard et al., 2016). Traditionally the ST systems are built by cascading ASR and MT systems (Ney, 1999). However, the cascaded system suffers from error propagation, latency, and memory requirement issues. Although these issues can be addressed using end-to-end ST models, it is hard to collect such data for training these models.

In this work, we build an end-to-end ST system which not only addresses the issues of a cascaded system but also works with limited training data. The proposed system is fine-tuned towards IWSLT 2020 Offline Speech-Translation Task [1]. However, the proposed training strategies and the data augmentation techniques can be adopted into existing and future ST models. We adopt the meta-learning approach proposed for ST task (Indurthi et al., 2019) to train our system. The meta-learning based training approach not only allows us to leverage huge amounts of training data available in ASR and MT tasks but also helps to find a good initialization point for the target ST task.

We conduct several experiments involving ASR, MT, and ST corpora to test our model performance on the IWSLT 2020, MuST-C, and Europarl-ST English-German (En-De) ST tasks. Our experiments reveal that the proposed model trained using the meta-learning approach achieves significant performance gains over the model which only utilizes the ST data for training. Our model achieves 4.81, 5.37, and 8.46 BLEU score improvements on IWSLT test 2015, MuST-C test, Europarl-ST test sets compared to the models trained without using the meta-learning approach for training. Our best system attains 24.58, 27.51, and 27.61 BLEU scores on IWSLT test 2015, MuST-C test, and Europarl-ST test sets, respectively.

2 Model Architecture

We use the Transformer model as a base Sequence-to-Sequence (seq2seq) model to train the ASR, MT, and ST tasks. In this section, we describe briefly about the Transformer architecture and how it is adopted to ASR and ST tasks. In Section 2.2, we describe the meta-learning algorithm used to train our seq2seq model.

2.1 Base Architecture

A general seq2seq architecture (Sutskever et al., 2014) generates a target sequence $\boldsymbol{y} =$

* The two authors contributed equally to this paper

[1] The International Conference on Spoken Language Translation ACL - 17th IWSLT 2020

Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT), pages 73–79

https://doi.org/10.18653/v1/P17

$\{y_1, \cdots, y_n\}$ given a source sequence $\boldsymbol{x} = \{x_1, \cdots, x_m\}$ by modeling the conditional probability, $p(\boldsymbol{y}|\boldsymbol{x}, \theta)$. The MT task is one example of seq2seq problems where $\boldsymbol{x}$ represents the input sequence in the source language and $\boldsymbol{y}$ represents the translated output sequence in the target language.

The non-recurrent Transformer network (Vaswani et al., 2017) has been extensively used to solve general seq2seq problems, especially the MT task. The Transformer is based on an encoder-decoder architecture (Cho et al., 2014). The encoder and decoder blocks of the Transformer network are composed of stacks of N, M identical layers. Each encoder layer has two sub-layers, the first being a multi-head self-attention mechanism, and the second sub-layer being a position-wise fully connected feed-forward network. Similarly, each decoder has these two sub-layers. In addition to these two sub-layers, the decoder contains an additional sub-layer for computing the encoder-decoder attention vector based on soft attention mechanism (Bahdanau et al., 2015).

2.2 MAML

Meta-Learning approach is proven to be very useful to mitigate the data scarcity issue in low resource tasks. Due to the scarcity of ST data in our task, we use the variant of meta-learning approach called Modality Agnostic Meta-Learning (MAML) (Finn et al., 2017a) to leverage high resource tasks when training on low resource tasks. Here, we briefly describe the MAML approach for the ST task. For more details about the meta-learning approach for the ST task, please refer to (Indurthi et al., 2019).

The MAML approach involves two phases: (1) Meta-Learning Phase, (2) Fine-tuning Phase. In the meta-learning phase, we use a set of related high resource tasks as source tasks to train the model. In this phase, the model captures the general learning aspects of the tasks involved. During the fine-tuning phase, we tune the model towards the specific target task after initializing the model from the parameters learned in the meta-learning phase.

Meta-Learning Phase: In this phase, we use the high resource tasks as source tasks $\{\tau^1, \cdots, \tau^s\}$ to find a good parameter initialization point θ^0 for the low resource target task τ^0. For each step in this phase, we first uniformly sample one source task τ at random from the set of source tasks $\{\tau^1, \cdots, \tau^s\}$. We then sample two batches(D_τ and $D_\tau^{'}$) of training examples from this task τ. The D_τ is used to train the model to learn the task specific distribution and this step is called meta-train step. In each meta-train step, we create auxiliary parameters (θ_τ^a) initialized from the original model parameters (θ^m). We update the auxiliary parameters during this step using D_τ while keeping original parameters intact. The auxiliary parameters (θ^a) are updated using the gradient-decent step and it is given by,

$$\theta_\tau^a = \theta^m - \alpha \nabla_{\theta^m} \ell(D_\tau; \theta^m). \quad (1)$$

After the meta-train step, the auxiliary parameters (θ^a) are evaluated on $D_\tau^{'}$ to compute the loss. This step is called meta-test and the computed loss is used to update the original model parameters (θ^m).

$$\theta_\tau^m = \theta^m - \beta \nabla_{\theta^a} \ell(D_\tau^{'}; \theta^a).` \quad (2)$$

Note that the meta-test step is performed over the model parameters (θ^m), whereas the loss is computed using the auxiliary parameters (θ^a). In effect, the meta-learning phase aims to optimize the model parameters such that a new low resource target task can be quickly learned during the fine-tuning phase.

Fine-tuning Phase: During fine-tuning phase, the model is initialized from the meta-learned parameters (θ^m) and trained on specific target task. In this phase, the model training is done like a usual neural network training without involving the auxiliary parameters.

Exposing the model parameters to vast amounts of data from high resource source tasks $\{\tau^1, \cdots, \tau^s\}$ during the meta-learning phase makes them suitable to act as a good initialization point for the target task τ^0.

2.3 Speech-to-text Translation:

We adopt the basic Transformer (Vaswani et al., 2017) architecture described in Section 2.1 to train ASR and ST tasks. We represent the speech sequence in these tasks using the Log Mel 80-dimensional features. The speech sequences are usually a few times longer than the text sequences. Thus, we add a compression layer at the beginning of the Transformer network to compress and extract structure locality from the speech sequences. This compressed signal is given as input to the Transformer encoder. The compression layer comprises

of a stack of CNN layers. The text sequences in all the ASR, MT, and ST tasks are represented using word piece vocabulary.

The limited amount of training data in the ST task can result in over-fitting and leads to an inferior performance. Hence, we use the meta-learning approach described in the Section 2.2. The meta-learning approach for ST task proposed by (Indurthi et al., 2019) suggests high resource tasks such as Automatic Speech Recognition (ASR) and Machine Translation (MT) as source tasks during meta-learning phase. Unlike (Indurthi et al., 2019), we include ST task as one of the source tasks during the meta-learning phase to leverage the ST training data as well. So, the set of source tasks in our meta-learning phase are $\{ASR, MT, ST\}$ and the target task τ^0 during the fine-tuning phase is ST. We dynamically disable the compression layer whenever we sample the MT task during the meta-learning phase. This allows us to train the model on the tasks with different input-output modalities.

During the meta-learning phase, the parameters of the model (θ^m) are exposed to vast amounts of speech-to-transcripts and text-to-text translation examples via ASR and MT tasks along with the original ST tasks' speech-to-text translation examples. This allows the parameters of all the sub-layers in the model such as compression, encoder, decoder, encoder-decoder attention, and output layers to learn the individual language representations and translation relations between them.

2.4 Training

The speech-to-text translation models are trained on a dataset D of parallel sequences to maximize the the log likelihood:

$$\ell(D;\theta) = \frac{1}{|D|}\sum_{i=1}^{|D|} \log p\left(\boldsymbol{y}^i|\boldsymbol{x}^i;\theta\right) \qquad (3)$$

where θ denotes the parameters of the model. To facilitate the training on multiple languages and tasks, we create a universal vocabulary by following (Gu et al., 2018). The universal vocabulary is created based on all the tasks involved in the meta-learning and fine-tuning phases.

3 Datasets

3.1 Dataset composition

Datasets used to train our model come from three different tasks, ASR, MT, and ST. All of these datasets are used during the meta-learning phase, while only the ST task dataset is used for fine-tuning. All the corpora we used are from the IWSLT 2020's allowed training data. The details of all the datasets are given in the Table 1.

Task	Corpus	# hours	# Examples
MT	Open Subtitles	N/A	22,512,639
MT	WMT 19	N/A	4,592,289
ASR	LibriSpeech	982	232,958
ASR	IWSLT 19 ST(filtered)	220	145,372
ASR	MuST-C	400	229,702
ASR	TED-LIUM 3	452	286,263
ST	Europarl-ST	89	32,628
ST	IWSLT 19 ST(filtered)	220	145,372
ST	MuST-C	400	229,703

Table 1: Number of original training examples in each dataset.

ST Task: For ST task, we used Europarl-ST(Iranzo-Sánchez et al., 2019), IWSLT 19(filtered), and MuST-C(Di Gangi et al., 2019) datasets. The total number of examples from these three datasets is 407K, where as the size of the ASR corpora is 894K examples. To resolve the ST data scarcity issue, we augment the training data for ST with various approaches described in the Section 3.2. Thus, we increased the size of the ST training data from 407K examples to 2.2M examples.

ASR Task: We used four different datasets to train the ASR English task, IWSLT 19(filtered), LibriSpeech(Panayotov et al., 2015), MuST-C, and TED-LIUM 3(Hernandez et al., 2018), which adds a total of 894K English speech-to-text transcripts. Although, IWSLT 19(filtered), MuST-C, and TED-LIUM 3 are ST corpora, they also have the English transcripts, so we include them into ASR tasks as well. We do not augment the ASR datasets with synthetic data, unlike the ST datasets. Adding more synthetic data for ASR task may bias the model towards ASR task rather than target ST task.

MT Task: WMT 19 and Open Subtitles(Lison et al., 2019) corpora are used for the MT task. The examples used for training MT come from Common Crawls, Europarl v9, and News Commentary v14 sets of WMT 19, which amounts to 27M training examples.

3.2 Data augmentation

For the data augmentation on the text side, we use two English-to-German NMT model and top-2 beam results to generate synthetic German sequences from the corresponding English sequences.

Corpus	Use original data	Speech Augmentation	Text Augmentation	# Pairs	# Examples
Europarl-ST	Y	$\times 2$	$\times 2$	3 pairs	97,884
IWSLT 19 ST(filtered)	Y	$\times 4$	$\times 4$	5 pairs	726,380
MuST-C	Y	$\times 3$	None	4 pairs	918,812
TED-LIUM 3	N	$\times 2$	$\times 2$	2 pairs	536,526

Table 2: Data augmentation strategies for the ST task.

For speech sequence, we use the Sox library to generate the speech signal using different values of speed, echo, and tempo parameters similar to (Potapczyk et al., 2019). The parameter values are uniformly sampled using these ranges : tempo $\in (0.85, 1.3)$, speed $\in (0.95, 1.05)$, echo_delay $\in (20, 200)$, and echo_decay $\in (0.05, 0.2)$. We increase the size of the IWSLT 19(filtered) ST dataset to five times of the original size by augmenting 4X data – four text sequences using the NMT models and four speech signals using the Sox parameter ranges. For the Europarl-ST, we augment 2X examples to triple the size. The TED-LIUM 3 dataset does not contain speech-to-text translation examples originally, hence, we create 2X synthetic speech-to-text translations using speech-to-text transcripts. Finally, for the MuST-C datasest, we use synthetic speech to increase the dataset size to 4X. Overall, we created the synthetic training data of size roughly equal to four times the original data using data augmentation techniques described above. The details of these synthetic datasets are given in the Table 2. During training, we also tried SpecAugment(Park et al., 2019) to increase the speech data, but it did not help to boost overall performance.

3.3 Data processing

In order to deal with different input and output modalities, we use universal vocabulary (Gu et al., 2018) generated from all the text data, i.e. ASR transcripts, MT source and target text and ST translations. For input speech signal in ASR and ST tasks, we use Log Mel 80-dimensional features to process the input speech. Additionally, to remove noisy data in IWSLT 19 ST dataset, we use a pre-trained ASR model to filter examples with word error rate (WER) ≥ 70.

Dev/Test set	# Examples
IWSLT Test 2010	1,568
IWSLT Test 2015	1,080
MuST-C Dev	1,423
MuST-C Test	2,641
Europarl-ST Dev	1,320
Europarl-ST Test	1,253

Table 3: The number of examples of dev and test sets.

4 Experiments

4.1 Implementation Details

We trained all our models on 4*NVIDIA V100 GPUs. The MAML model is implemented based on the Tensor2Tensor framework (Vaswani et al., 2018). We train the models in the meta-learning phase for 1600k steps and then finetune for 400k steps. The compression layer is composed of three CNN layers. The number of encoder and decoder layers(N and M) in the base transformer model is set to 10 and 8, respectively. In all the experiments, a dropout rate of 0.2 is used. We use a batch size of 1.5M frames for the speech sequences and a batch size of 4096 tokens for the text sequences. In order to deal with small batches due to long speech signals, we use Multistep Adam optimizer (Saunders et al., 2018) in our experiments, with the gradients accumulated over 32 steps.

4.2 Results

In this section, we report the performance of our models on different ST datasets. We report the performance of models on IWSLT tst 2010, tst 2015, MuST-C dev, MuST-C test, Europarl-ST dev, and Europarl-ST test sets. The number of examples in these test sets are reported in the Table 3.

We trained one model using only ST datasets shown in Table 2, called *woML* (without Meta-Learn) from here on. This model *woML* is trained without using the meta-learning approach. We trained another model, called *wML* (with Meta-

Model	IWSLT		MuST-C		Europarl-ST	
	tst 2010	tst 2015	Dev	Test	Dev	Test
woML (without Meta-Learn)	20.21	19.77	16.8	22.14	19.23	19.15
wML (with Meta-Learn)	25.98	24.4	22	26.77	25.8	26.8
Model Averaging	**26.43**	**24.58**	**23.59**	**27.51**	**26.88**	**27.61**

Table 4: Performance of models trained using with/without meta-learning approach on various datasets.

Learn), in which we first pre-train the model using the meta-learning approach described in the Section 2.2 using all the ASR, MT, ST tasks. We then finetune the model from the meta-learned parameters on the ST task. As we can see from the Table 4, the *wML* model achieves a better BLEU score than *woML* on all the ST datasets. We see that the *wML* model out-performs *woML* by achieving a BLEU score of 24.4 on IWSLT 2015 test set as compared to the 19.77 BLEU score achieved by *woML*. These results clearly show that the meta-learning phase helps to leverage the data from ASR, MT datasets and helps to learn the individual language representations and the relations between them.

We got further improvements on the ST BLEU score by averaging 10 checkpoints around the best model. In the Table 4, one can see that ensemble model attained an improvement of 0.18 BLEU score on IWSLT 2015 test set, 0.74 BLEU score on MuST-C test set, 0.81 BLEU score on Europarl-ST test sets. The ensemble model achieved a performance of 24.58 BLEU score on IWSLT 2015 test set by using meta-learning, data augmentation and average checkpoint techniques.

5 Related Work

End-to-end Speech Translation: Previously, speech translation leveraged the success of MT and ASR systems to build the cascade speech translation system(Post et al., 2013). The cascade models mostly suffer from problems such as propagating errors between models and high latency during decoding. In order to overcome these limitations, various attempts have been made to develop end-to-end ST models by aligning source speech signal and target text translation without using intermediate transcripts(Duong et al., 2016). However, due to the limited availability of training data unlike ASR or MT corpora, various data augmentation strategies have been proposed to leverage the data from ASR or MT tasks to improve the end-to-end ST(Jia et al., 2019; Pino et al., 2019) performance. Recently, several learning approaches such as multi-task learning using either ASR+ST or MT+ST data pairs have been suggested and explored. However, in these approaches, the parameters of the model are updated independently based on individual task performance, which may lead to sub-optimal solutions. Indurthi et al. (2019) proposed a meta-learning approach to overcome these limitations.

Meta-Learning: Meta-learning algorithms are used to adapt quickly to new tasks with relatively few examples as the main goal of the algorithm is learning to learn. Unlike the past meta-learning approaches which focused on learning a meta policy(Ha et al., 2016; Andrychowicz et al., 2016), (Finn et al., 2017b) recently proposed a meta-learning algorithm which puts more weight on finding a good initialization point for new target tasks.

6 Conclusion

In this work, we improve the performance of end-to-end speech translation system based on the data available from the IWSLT2020 Offline Speech Translation Task. We train end-to-end models to solve the complex task of speech translation. We leverage the large out-of-domain training data from the ASR, MT tasks to improve the performance of the ST task. We adopt Model Agnostic Meta-Learning(MAML) and data augmentation techniques to achieve a performance of 24.58, 27.51, 27.61 BLEU scores on IWSLT test 2015, MuST-C test, and Europarl-ST test sets respectively.

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *Advances in neural information processing systems*, pages 3981–3989.

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondrej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco

Turchi, and Changhan Wang. 2020. Findings of the IWSLT 2020 Evaluation Campaign. In *Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT 2020)*, Seattle, USA.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate.

Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. 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)*, pages 2012–2017, Minneapolis, Minnesota. Association for Computational Linguistics.

Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn. 2016. An attentional model for speech translation without transcription. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 949–959.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017a. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017b. Model-agnostic meta-learning for fast adaptation of deep networks.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 38.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 344–354, New Orleans, Louisiana. Association for Computational Linguistics.

David Ha, Andrew Dai, and Quoc V Le. 2016. Hypernetworks. *arXiv preprint arXiv:1609.09106*.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. Ted-lium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. *Lecture Notes in Computer Science*, page 198–208.

Sathish Indurthi, Houjeung Han, Nikhil Kumar Lakumarapu, Beomseok Lee, Insoo Chung, Sangha Kim, and Chanwoo Kim. 2019. Data efficient direct speech-to-text translation with modality agnostic meta-learning. *arXiv preprint arXiv:1911.04283*.

Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan. 2019. Europarl-st: A multilingual corpus for speech translation of parliamentary debates. *arXiv preprint arXiv:1911.03167*.

Ye Jia, Melvin Johnson, Wolfgang Macherey, Ron J Weiss, Yuan Cao, Chung-Cheng Chiu, Naveen Ari, Stella Laurenzo, and Yonghui Wu. 2019. 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)*, pages 7180–7184. IEEE.

Pierre Lison, Jörg Tiedemann, Milen Kouylekov, et al. 2019. Open subtitles 2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *LREC 2018, Eleventh International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA).

Hermann Ney. 1999. Speech translation: Coupling of recognition and translation. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 1, pages 517–520. IEEE.

V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *Interspeech 2019*.

Juan Pino, Liezl Puzon, Jiatao Gu, Xutai Ma, Arya D McCarthy, and Deepak Gopinath. 2019. 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)*.

Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. 2013. Improved speech-to-text translation with the fisher and callhome spanish–english speech translation corpus. In *Proc. IWSLT*.

Tomasz Potapczyk, Pawel Przybysz, Marcin Chochowski, and Artur Szumaczuk. 2019. Samsung's system for the iwslt 2019 end-to-end speech translation task. Zenodo.

Danielle Saunders, Felix Stahlberg, Adrià de Gispert, and Bill Byrne. 2018. Multi-representation ensembles and delayed SGD updates improve syntax-based NMT. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 319–325, Melbourne, Australia. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan N. Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, Ryan Sepassi, Noam Shazeer, and Jakob Uszkoreit. 2018. Tensor2tensor for neural machine translation. *CoRR*, abs/1803.07416.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.