On Knowledge Distillation for Translating Erroneous Speech Transcriptions

Ryo Fukuda[1], **Katsuhito Sudoh**[1,2], **and Satoshi Nakamura**[1,2]
[1]Nara Institute of Science and Technology, Japan
[2]AIP, RIKEN, Japan
{fukuda.ryo.fo3, sudoh, s-nakamura}@is.naist.jp

Abstract

Recent studies argue that knowledge distillation is promising for speech translation (ST) using end-to-end models. In this work, we investigate the effect of knowledge distillation with a cascade ST using automatic speech recognition (ASR) and machine translation (MT) models. We distill knowledge from a teacher model based on human transcripts to a student model based on erroneous transcriptions. Our experimental results demonstrated that knowledge distillation is beneficial for a cascade ST. Further investigation that combined knowledge distillation and fine-tuning revealed that the combination consistently improved two language pairs: English-Italian and Spanish-English.

1 Introduction

Speech translation (ST) converts utterances in a source language into text in another language. Conventional ST systems called *cascade* or *pipeline* ST consist of two components: automatic speech recognition (ASR) and machine translation (MT). In the cascade ST, the error propagation from ASR to MT seriously degrades the ST performance. On the other hand, a new ST system called *end-to-end* or *direct* ST uses a single model to directly translate the source language speech into target language text (Bérard et al., 2016). Such an end-to-end approach is a new paradigm in ST and is attracting much research attention. However, a naive end-to-end ST without additional training, such as ASR tasks, remains inferior to a cascade ST (Liu et al., 2018; Salesky and Black, 2020). Additionally, it requires parallel data of the source language speech and the target language text, which cannot be obtained easily in practice.

Recent ST studies have incorporated the techniques of cascade ST to end-to-end STs. Multitask training with an ASR subtask has been used successfully in end-to-end ST (Weiss et al., 2017; Anastasopoulos and Chiang, 2018; Sperber et al., 2019). Initializing an end-to-end ST with a pretrained ASR or MT has also become a common approach (Bérard et al., 2018; Bansal et al., 2019; Inaguma et al., 2020; Wang et al., 2020; Bahar et al., 2021).

In this work, we focus on the cascade approach due to its performance advantage against end-to-end STs. Another reason is that cascade ST models can be incorporated into end-to-end STs, as shown in previous studies.

During the training of an MT model for a cascade ST, we can use clean human transcripts for the source language speech as input. However, since the MT in a cascade ST always receives ASR output during inferences, ASR errors should be propagated to the MT model to cause translation errors. What if we use erroneous speech transcriptions by ASR for training? That approach means the MT model is trained to translate *erroneous* transcriptions into *correct* text, which would not generally be appropriate. One possible solution is to use both types of input (clean and erroneous transcriptions) for training, not just one. The question is how to use them. What is the proper training strategy for cascade STs? This is what we want to learn.

In this work, we address such problems by applying knowledge distillation to cascade STs. We distill the knowledge of a teacher model based on clean transcriptions to a student model based on erroneous transcriptions. We also investigate the joint use of knowledge distillation and fine-tuning. Experimental results revealed that the knowledge distillation improved the robustness against ASR errors and that the knowledge distillation after the fine-tuning provided more significant improvement.

Proceedings of the 18th International Conference on Spoken Language Translation, pages 198–205
Bangkok, Thailand (Online), August 5–6, 2021.

2 Related work

Some ST studies have tackled the problem of ASR error propagation. N-best hypotheses (Zhang et al., 2004; Quan et al., 2005), confusion networks (Bertoldi and Federico, 2005; Bertoldi et al., 2007), and lattices (Matusov and Ney, 2010; Sperber et al., 2017a) were used to include ASR ambiguity in the ST process.

Osamura et al. (2018) used the weighted sum of embedding vectors for ASR word hypotheses based on their posterior probabilities. Sperber et al. (2017b) and Xue et al. (2020) showed that translation accuracy against erroneous speech transcriptions can be improved by introducing pseudo ASR errors in the training data of MT.

Knowledge distillation (KD) (Buciluǎ et al., 2006; Hinton et al., 2015) is a method of transferring knowledge from a teacher to a student model. Typically, the student model is trained by minimizing the KL-divergence (Kullback and Leibler, 1951) loss between the output probability distributions of the teacher and student models (word-level KD). Sequence-level knowledge distillation (sequence-level KD) (Kim and Rush, 2016a) targets the token-sequence generated by the teacher model using beam search. In our experiments, sequence-level KD outperformed word-level one, and Kim and Rush (2016b) showed similar trends. Therefore, in our experiments, we call it KD.

The KD technique is prevalent in many applications of machine learning, including MT (non-autoregressive machine translation (Gu et al., 2017), simultaneous translation (Ren et al., 2020), etc.). Typically, it is used to distill knowledge from a larger teacher model to a smaller or faster student model. Recent works (Furlanello et al., 2018; Yang et al., 2018) have shown that the student model's accuracy exceeds that of the teacher model, even if its size is identical as the student model. KD has also been applied to ST. Gaido et al. (2020) applied KD to an end-to-end ST using an MT model based on clean transcriptions as the teacher of the end-to-end ST model. Our work focuses on the application of KD to a cascade ST using a teacher model based on clean transcripts for the student model that takes erroneous inputs.

Dakwale and Monz (2019) proposed distillation as a remedy for the effective use of noisy parallel data for machine translation. They first trained the teacher model only on high-quality, clean data. Then they fed the source-side of the noisy parallel data into the teacher model and trained the student model to translate from the noisy source to the teacher's output. The main difference between their work and ours is that we have loosely equivalent source sentences (clean or erroneous transcription), which can be paired with the same target sentence. Therefore, the student model can be trained with more reliable objectives obtained by feeding clean transcriptions to the teacher model.

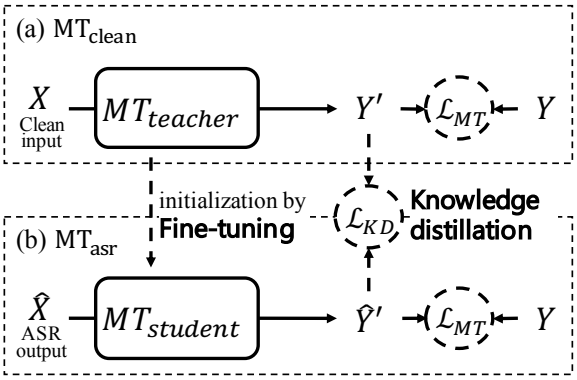

Figure 1: Overview of key concepts of methods

3 Cascade ST

Suppose triplet $W = (w_1, ..., w_J)$, $X = (x_1, ..., x_K)$, and $Y = (y_1, ..., y_L)$, where $W(1 \leq j \leq J)$, $X(1 \leq k \leq K)$, and $Y(1 \leq l \leq L)$ are sequences of the speech features in a source language, the corresponding transcribed source language tokens, and translated target language tokens.

In a cascade ST, first the ASR model is trained by the W and X pair. Then the MT model is trained to translate from X to Y. The loss function of MT model $\mathcal{L}_{MT}$ is defined using cross entropy:

$$\mathcal{L}_{MT} = -\sum_{l=1}^{L}\sum_{v\in V}^{|V|} logP(y_l = v), \quad (1)$$

where $P(y_l = v)$ is the posterior probability of candidate v in target language vocabulary V at time l in Y:

$$P(y_l = v) = p(v|X, y_{<l}; \theta). \quad (2)$$

4 Proposed method

When training an MT model, we can also use $\hat{X}$ instead of X, which is the output of the ASR model. We call the model trained with clean input X $\mathrm{MT}_{\mathrm{clean}}$ (Fig. 1(a)) and the one trained with ASR-based input $\hat{X}$ $\mathrm{MT}_{\mathrm{asr}}$ (Fig. 1(b)).

4.1 Joint use of KD and FT

To most effectively exploit both clean input X and ASR-based input $\hat{X}$, we introduce two training techniques: KD and fine-tuning. In KD, the student model is trained using $\hat{X}$ by minimizing loss $\mathcal{L}_{KD}$. As shown in Fig. 1, $\mathcal{L}_{KD}$ is the loss between $Y' = (y_1^{'}, ..., y_M^{'})$ and $\hat{Y} = (\hat{y_1}, ..., \hat{y_N})$, where $Y'(1 \leq m \leq M)$ and $\hat{Y}(1 \leq n \leq N)$ are the outputs of the teacher and student models. We use the sequence-level KD so that $\mathcal{L}_{KD}$ is calculated by replacing L with M and l with m in Eq. 1.

On the other hand, fine-tuning (FT) has been widely used for domain adaptation in MT (Sennrich et al., 2016a). Di Gangi et al. (2019c) showed that a model fine-tuned with ASR-based input becomes robust to erroneous ASR input while maintaining high performance for clean input. Following this finding, we employ FT for MT training. In FT, the student model with $\hat{X}$, which inherits the parameters of the teacher model with X, is trained by minimizing $\mathcal{L}_{MT}$ (Fig. 1).

In addition to the independent use of KD and FT, we examined their possible combinations:

- **FT+KD**. Apply these techniques at the same time. Unlike regular FT, we use loss $\mathcal{L}_{KD}$ instead of $\mathcal{L}_{MT}$.
  Specifically, (1) the teacher model is trained with clean input X and loss $\mathcal{L}_{MT}$. Then (2) the student model is trained with ASR-based input $\hat{X}$ and loss $\mathcal{L}_{KD}$, inheriting the parameters of the teacher model.

- **KD→FT**. Perform additional training with $\mathcal{L}_{MT}$ to the model trained by KD.
  Specifically, (1) the student model is trained with $\hat{X}$ and $\mathcal{L}_{KD}$. Then (2) fine-tune the model with $\hat{X}$ and $\mathcal{L}_{MT}$.

- **FT→KD**. Perform additional training with $\mathcal{L}_{KD}$ to the model trained by FT.
  Specifically, (1) the student model is trained with $\hat{X}$ and $\mathcal{L}_{MT}$, inheriting the parameters of the teacher model. Then (2) fine-tune the model with $\hat{X}$ and $\mathcal{L}_{KD}$.

5 Experiments

5.1 Dataset

We conducted experiments for English to Italian and Spanish to English NMT. For English-Italian, we used MuST-C (Di Gangi et al., 2019a), a multilingual ST corpus built from TED talks. It contains triplets of about 250K segments of English speeches, transcripts, and Italian translations. We used audio and transcript pairs to train the ASR. To train the MT model, we used transcripts as clean input and ASR outputs as noisy input.

For Spanish-English, we used LDC Fisher Spanish speech with new English translations (Post et al., 2013; Salesky et al., 2018). It has the following roughly 140K segments of multi-way parallel data:

1. Spanish disfluent speech
2. Spanish clean transcriptions
3. Spanish erroneous transcriptions (ASR output)
4. English disfluent translations
5. English fluent translations

When we train the MT model, we used (5) as output. For the sake of reproducibility we used (2) or (3) as clean or noisy input included in the dataset.

We preprocessed the text data with Byte Pair Encoding (BPE) (Sennrich et al., 2016b) to split the sentences into subwords. The vocabulary size was set to 8,000 in all the languages. For the English audio, we extracted 80-channel log mel filterbank features (25-ms window size and 10-ms shift) and applied an utterance-level CMVN.

To evaluate the performance, we calculated the case-sensitive BLEU with sacreBLEU.[1] We measured BLEU for both the ASR-based and clean input to evaluate the ASR error robustness and the topline performance in an ideal situation without ASR errors.

5.2 Model

We used the Transformer (Vaswani et al., 2017) implementation of Fairseq[2] to construct both the ASR and the MT. The hyper-parameters of the model generally follow the Transformer Base settings (Vaswani et al., 2017). Each encoder and decoder has 6 sub-layers. We set the word embedding dimensions, the hidden state dimensions, and the feed-forward dimensions to 512, 512, and 2,048. We performed the sub-layer's dropout with a probability of 0.1 and employed 8 attention heads for both the encoder and the decoder. The model is trained using Adam with an initial learning rate

[1] https://github.com/mjpost/sacreBLEU
[2] https://github.com/pytorch/fairseq

ST Type	System	ASR-based input	Clean input
End-to-end	ST + ASR-PT (Di Gangi et al., 2019b)[1]	16.8	
	ST + ASR-PT (*ESPnet*)[2]	21.5	
	ST	17.0	
	ST + ASR-PT	21.4	
Cascade	$\mathrm{MT}_{\mathrm{clean}}$ (Di Gangi et al., 2019b)[1]	18.9	-
	$\mathrm{MT}_{\mathrm{clean}}$	22.4	29.7
	$\mathrm{MT}_{\mathrm{asr}}$	22.1	27.2
	$\mathrm{MT}_{\mathrm{asr}}$ + FT	23.2	29.8
	$\mathrm{MT}_{\mathrm{asr}}$ + KD	22.5	28.2
	$\mathrm{MT}_{\mathrm{asr}}$ + FT + KD	23.4	29.9
	$\mathrm{MT}_{\mathrm{asr}}$ + KD $\rightarrow$ FT	23.1	29.3
	$\mathrm{MT}_{\mathrm{asr}}$ + FT $\rightarrow$ KD	**23.5**	**30.2**

Table 1: ST systems on MuST-C English-Italian. Test BLEU reported. [1]End-to-end (above) or cascade ST (below) systems using Fairseq's Transformer Base model, which resembles our conditions. [2]End-to-end ST system using ESPnet resembles our conditions chosen from a report (https://github.com/espnet/espnet/blob/master/egs/must_c/st1/RESULTS.md).

of 0.0007, $\beta_1 = 0.9$, and $\beta_2 = 0.98$, following Vaswani et al. (2017). We used 4,096 tokens per mini-batch and eight iterations of forward-passes, accumulated gradients, and back-propagated them. Validation was performed every 1,000 updates, and the test checkpoint with the best loss was stored.

For English-Italian, we also built several end-to-end ST variants using Fairseq for comparison with the cascade models. All the settings are identical as in MT: using Transformer described above and trained with label-smoothed cross entropy loss.

6 Results

6.1 English-Italian

Table 1 shows the BLEU results for the English to Italian NMT. In the end-to-end systems, a naive model (ST) without any additional technique, such as an ASR subtask, was significantly lower than the others and was significantly improved by pre-training the ASR encoder (ST + ASR-PT).

The cascade methods worked better than the end-to-end methods. In the cascade ST, the performance of a system trained using only ASR input ($\mathrm{MT}_{\mathrm{asr}}$) was worse (0.3-BLEU drop for the ASR-based test data and 2.5-BLEU drop for the clean test data) than the clean input ($\mathrm{MT}_{\mathrm{clean}}$). The ASR-based training data contained erroneous transcriptions of WER 14.49, leading to degradation. On the other hand, some systems trained using both ASR input and clean input were better than $\mathrm{MT}_{\mathrm{clean}}$ when translating clean input. This indicates that the training with ASR errors may contribute to regularize the model, which yields improvements.

The FT for the ASR-based input ($\mathrm{MT}_{\mathrm{asr}}$ + FT) showed improvements for the ASR-based input (+1.1 BLEU). Compared to FT, KD ($\mathrm{MT}_{\mathrm{asr}}$ + KD) produced a small improvement with the ASR-based input (+0.4 BLEU). In the KD, a teacher model got a BLEU score of 41.6 on the reference for training data.

With respect to the joint use of FT and KD, simultaneously applying these techniques ($\mathrm{MT}_{\mathrm{asr}}$ + FT + KD) shows only slight improvements (+0.2 BLEU for ASR-based test data and +0.1 BLEU for clean test data), compared to FT only ($\mathrm{MT}_{\mathrm{asr}}$ + FT). Applying FT after KD ($\mathrm{MT}_{\mathrm{asr}}$ + KD $\rightarrow$ FT) was inferior to the other combinations, especially for clean data, probably because the MT was not trained with clean input. Distilling knowledge after FT ($\mathrm{MT}_{\mathrm{asr}}$ + FT $\rightarrow$ KD) gave the best score for both the ASR-based and the clean test data. FT enables the student model to learn good parameter values, and KD provides the student model with its upper bounds from the teacher model.

6.2 Spanish-English

Table 2 shows the overall results for the Spanish to English cascade ST. They are similar to those in English-to-Italian; FT and KD improved BLEU, and combining them yielded more significant improvements. However, the gap was larger for the clean test data between systems only trained on the ASR-based input ($\mathrm{MT}_{\mathrm{asr}}$) and only on the clean input ($\mathrm{MT}_{\mathrm{clean}}$). The ASR-based training data contained many erroneous transcriptions of WER 36.5,

System	Fisher/Test 0		Fisher/Test 1	
	ASR-based input	Clean input	ASR-based input	Clean input
$\mathrm{MT}_{\mathrm{clean}}$	17.5	**26.8**	17.0	**26.1**
$\mathrm{MT}_{\mathrm{asr}}$	17.5	17.6	16.9	17.2
$\mathrm{MT}_{\mathrm{asr}} + \mathrm{FT}$	18.3	24.9	17.5	24.5
$\mathrm{MT}_{\mathrm{asr}} + \mathrm{KD}$	18.5	16.5	17.9	16.2
$\mathrm{MT}_{\mathrm{asr}} + \mathrm{FT} + \mathrm{KD}$	18.8	25.2	18.0	24.9
$\mathrm{MT}_{\mathrm{asr}} + \mathrm{KD} \rightarrow \mathrm{FT}$	17.8	15.7	17.1	15.3
$\mathrm{MT}_{\mathrm{asr}} + \mathrm{FT} \rightarrow \mathrm{KD}$	**19.0**	25.2	**18.4**	25.2

Table 2: ST systems on Fisher Spanish-English. Test BLEU for two fluent references reported.

causing more serious degradation. It also differs from the English-to-Italian experiments in that KD ($\mathrm{MT}_{\mathrm{asr}} + \mathrm{KD}$) was superior to FT ($\mathrm{MT}_{\mathrm{asr}} + \mathrm{FT}$) for the ASR-based test data when it was used alone. In KD, BLEU using the teacher model as training data was 48.0, which is higher than 41.6 for English-Italian. One possible reason is that there was a higher upper bound that can be trained by KD. Another difference was a gap between the clean and ASR-based inputs, which have many erroneous transcriptions of WER 36.5. In such a case, parameter initialization by FT may not be very helpful.

In spite of the differences between the two experiments, we achieved consistent improvement by combining FT and KD.

7 Discussion

We analyzed the results with the Spanish to English models to discuss how erroneous transcriptions affect translation results and how KD and FT work.

Erroneous transcription The example below shows the problem of error propagation:

- (Clean input) *uno super, super nuevo que salio*
- (ASR output) *en un sur super nuevo que salio*
- (Reference) *One super new that came out*
- ($\mathrm{MT}_{\mathrm{asr}}$ with ASR-based input) *In the South, it came out*
- ($\mathrm{MT}_{\mathrm{asr}} + \mathrm{KD}$ with ASR-based input) *In a super new one that came out.*

Here the Spanish word *super* was misrecognized as *sur* by the ASR. This error was propagated to MT, and $\mathrm{MT}_{\mathrm{asr}}$ translated it as *South*. Although the word's translation itself from *sur* to *South* was not wrong, but it is not what we wanted. The model trained by KD ignored this error and generated a more proper sentence.

We found such ASR error correction phenomena in the results, although KD and FT did not directly address this issue.

Effect of Knowledge Distillation Spoken language parallel data have translations of colloquial spoken utterances. They increase the difficulty of training MT. For instance:

- (Clean input) *le ayuda si si, no es, no es interesante pero entonces, a ba- entonces ya despues cuando eso termino, tiene que escribir varios asi, ensayos, hacer un analisis*
- (Reference) *You have to write some essays like that, to make an analysis*
- (KD teacher) *It helps her yes, it's not interesting but then, when I finish, you have to write several, you have to make an analysis*

A human translator ignored many disfluent utterances from the original text, resulting in low fidelity. Here are some other examples:

- Inconsistent translations: "*De Venezuela*" was translated into "*From Venezuela*" at one time and "*Venezuela?*" at another time.
- All-caps: "*donde hay problemas*" was capitalized and translated into "*WHEN TROUBLE ARISES.*"
- Omission of a part of speech: "*Porque, tengo el, el bodysuit, pero*" was translated into "*I have the bodysuit.*." The conjunction "*pero* (but)" was removed for fluency.

The MT model can be confused by such translations. KD forces the student model to mimic literal teacher translations that may include some errors instead of reproducing translations of colloquial spoken utterances.

Effect of Fine-tuning Sometimes the fine-tuned MT model corrected the ASR errors:

- (Clean input) *Eh, para mi pues, eh, tengo como diez mil canciones en, en el, en la Ipod*
- (ASR output) *eh para mi pues eh tengo como diez mil canciones en en la epod*
- (Reference) *I have ten thousand songs in the Ipod.*
- ($\mathrm{MT}_{\mathrm{clean}}$ with ASR-based input) *To me, I have about ten thousand songs in the ethics*
- ($\mathrm{MT}_{\mathrm{asr}} + \mathrm{FT}$ with ASR-based input) *I have about ten thousand songs in the Ipod*

The ASR misrecognized "*Ipod*" as "*epod*," and the model before FT, which was only trained with clean inputs, incorrectly translated it as "*ethics*." As a result of the FT with ASR-based inputs, the model successfully translated it as "*Ipod*." The FT for the erroneous ASR outputs may have provided robustness against common errors.

8 Conclusion

We presented and discussed the benefits of using two machine learning techniques in cascade ST: knowledge distillation and fine-tuning. Our experimental results showed the advantages of the proposed method in two different conditions. Our results also suggest that combining knowledge distillation and fine-tuning is more beneficial than using either one because they have different roles.

In future work, we will incorporate our findings into an end-to-end ST to grow speech translation.

Acknowledgements

Part of this work was supported by JSPS KAKENHI Grant Number JP17H06101.

References

Antonios Anastasopoulos and David Chiang. 2018. Tied multitask learning for neural speech translation. 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 82–91, New Orleans, Louisiana. Association for Computational Linguistics.

Parnia Bahar, Tobias Bieschke, Ralf Schlüter, and Hermann Ney. 2021. Tight integrated end-to-end training for cascaded speech translation. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 950–957. IEEE.

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

Alexandre Bérard, Laurent Besacier, Ali Can Kocabiyikoglu, and Olivier Pietquin. 2018. End-to-end automatic speech translation of audiobooks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6224–6228. IEEE.

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*.

Nicola Bertoldi and Marcello Federico. 2005. A new decoder for spoken language translation based on confusion networks. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.*, pages 86–91. IEEE.

Nicola Bertoldi, Richard Zens, and Marcello Federico. 2007. Speech translation by confusion network decoding. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–1297. IEEE.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541.

Praveen Dakwale and Christof Monz. 2019. Improving neural machine translation using noisy parallel data through distillation. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 118–127.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019a. 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.

Mattia A Di Gangi, Matteo Negri, and Marco Turchi. 2019b. Adapting transformer to end-to-end spoken language translation. In *INTERSPEECH 2019*, pages 1133–1137. International Speech Communication Association (ISCA).

Mattia Antonino Di Gangi, Enyedi Robert, Brusadin Alessandra, and Marcello Federico. 2019c. Robust neural machine translation for clean and noisy speech transcripts. In *16th International Workshop on Spoken Language Translation 2019*.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born again neural networks. In *International Conference on Machine Learning*, pages 1607–1616. PMLR.

Marco Gaido, Mattia A. Di Gangi, Matteo Negri, and Marco Turchi. 2020. On knowledge distillation for direct speech translation.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor OK Li, and Richard Socher. 2017. Non-autoregressive neural machine translation. *arXiv preprint arXiv:1711.02281*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hirofumi Inaguma, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Yalta, Tomoki Hayashi, and Shinji Watanabe. 2020. ESPnet-ST: All-in-one speech translation toolkit. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 302–311, Online. Association for Computational Linguistics.

Yoon Kim and Alexander M Rush. 2016a. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*.

Yoon Kim and Alexander M. Rush. 2016b. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas. Association for Computational Linguistics.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Dan Liu, Junhua Liu, Wu Guo, Shifu Xiong, Zhiqiang Ma, Rui Song, Chongliang Wu, and Quan Liu. 2018. The ustc-nel speech translation system at iwslt 2018. *arXiv e-prints*, pages arXiv–1812.

Evgeny Matusov and Hermann Ney. 2010. Lattice-based asr-mt interface for speech translation. *IEEE transactions on audio, speech, and language processing*, 19(4):721–732.

Kaho Osamura, Takatomo Kano, Sakriani Sakti, Katsuhito Sudoh, and Satoshi Nakamura. 2018. Using spoken word posterior features in neural machine translation. *architecture*, 21:22.

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 *International Workshop on Spoken Language Translation*.

Vu Hai Quan, Marcello Federico, and Mauro Cettolo. 2005. Integrated n-best re-ranking for spoken language translation. In *Ninth European Conference on Speech Communication and Technology*.

Yi Ren, Jinglin Liu, Xu Tan, Chen Zhang, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2020. SimulSpeech: End-to-end simultaneous speech to text translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3787–3796, Online. Association for Computational Linguistics.

Elizabeth Salesky and Alan W Black. 2020. Phone features improve speech translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2388–2397, Online. Association for Computational Linguistics.

Elizabeth Salesky, Susanne Burger, Jan Niehues, and Alex Waibel. 2018. Towards fluent translations from disfluent speech. *2018 IEEE Spoken Language Technology Workshop (SLT)*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. 2017a. Neural lattice-to-sequence models for uncertain inputs. *arXiv preprint arXiv:1704.00559*.

Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. 2019. Attention-passing models for robust and data-efficient end-to-end speech translation. *Transactions of the Association for Computational Linguistics*, 7:313–325.

Matthias Sperber, Jan Niehues, and Alex Waibel. 2017b. Toward robust neural machine translation for noisy input sequences. In *International Workshop on Spoken Language Translation (IWSLT)*.

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.

Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Dmytro Okhonko, and Juan Pino. 2020. Fairseq S2T: Fast speech-to-text modeling with fairseq. In *Proceedings of the 1st Conference of the Asia-Pacific*

Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations, pages 33–39, Suzhou, China. Association for Computational Linguistics.

Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. *Proc. Interspeech 2017*, pages 2625–2629.

Haiyang Xue, Yang Feng, Shuhao Gu, and Wei Chen. 2020. Robust neural machine translation with asr errors. In *Proceedings of the First Workshop on Automatic Simultaneous Translation*, pages 15–23.

Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan Yuille. 2018. Knowledge distillation in generations: More tolerant teachers educate better students. *arXiv preprint arXiv:1805.05551*.

Ruiqiang Zhang, Genichiro Kikui, Hirofumi Yamamoto, Frank K Soong, Taro Watanabe, and Wai-Kit Lo. 2004. A unified approach in speech-to-speech translation: integrating features of speech recognition and machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1168–1174.