Amazon Alexa AI's System for IWSLT 2022 Offline Speech Translation Shared Task

Akshaya Vishnu Kudlu Shanbhogue* **Ran Xue*** **Ching-Yun Chang** **Sarah Campbell**
Amazon Alexa AI
{ashanbho,ranxue,cychang,srh}@amazon.com

Abstract

This paper describes Amazon Alexa AI's submission to the IWSLT 2022 Offline Speech Translation Task. Our system is an end-to-end speech translation model that leverages pretrained models and cross modality transfer learning. We detail two improvements to the knowledge transfer schema. First, we implemented a new loss function that reduces knowledge gap between audio and text modalities in translation task effectively. Second, we investigate multiple finetuning strategies including sampling loss, language grouping and domain adaption. These strategies aims to bridge the gaps between speech and text translation tasks. We also implement a multi-stage segmentation and merging strategy that yields improvements on the unsegmented development datasets. Results show that the proposed loss function consistently improves BLEU scores on the development datasets for both English-German and multilingual models. Additionally, certain language pairs see BLEU score improvements with specific finetuning strategies.

1 Introduction

Multilingual Spoken Language Translation (SLT) enables translation of audio into text in multiple languages. Traditionally, SLT is solved by cascading automatic speech recognition (ASR) models, which convert audio to transcribed text, with text-to-text translation models. End-to-end (E2E) models, such as FAIR Speech Translation System (Tang et al., 2021a), allow a single model to translate from speech to text. Recent advances in E2E models show comparable results with cascaded architectures (Anastasopoulos et al., 2021; Ansari et al., 2020).

Our baseline end-to-end speech translation system leverages large-scale pretrained models on different data modalities following the approach proposed by Tang et al. (2021a). We adopt dynamic dual skew divergence (DDSD) loss function (Li et al., 2021b) to replace cross entropy (CE) for effective knowledge transfer from pretrained text-to-text (T2T) translation model to speech-to-text (S2T) translation model through joint task training. We observe that DDSD consistently outperforms CE across all language directions.

Our multilingual model supports translation of English (en) audio to German (de), Japanese (ja) and Chinese (zh). We find that finetuning this model based on language groups can improve the performance of the model. Additionally, we find that finetuning models by considering alternate translations can lead to subtle improvements in the overall performance of the models. While working with unsegmented data, we show that using a custom audio segmentation strategy can improve the translation performance by around +2.0 BLEU points. On IWSLT 2022 blind test sets, our system achieves 22.6, 15.3, and 30.4 BLEU score for en→de, en→ja, and en→zh respectively. On the progression test set, our E2E speech translation system performs on par with IWSLT 2021 winning cascaded system (Anastasopoulos et al., 2021).

2 Base Model

We adopt the end-to-end speech translation system proposed by Tang et al. (2021a), which takes both text and speech as input for translation task. The model's encoder consists of a text encoder and a speech encoder for each input data modality, respectively. The text encoder is a 12 layer transformer architecture initialized from the pretrained mBART encoder (Tang et al., 2020). The speech encoder is a 24 layer transformer architecture in which we initialize the speech feature extractor and first 12 layers from pretrained Wav2Vec 2.0 model (Xu et al., 2020). The remaining 12 layers of the speech encoder share weights with the text encoder.

*Akshaya Vishnu Kudlu Shanbhogue and Ran Xue have equal contribution to this work.

Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022), pages 169 - 176

Between the speech encoder and text encoder, an adaptor (Li et al., 2021a) of 3 1-D convolution layers with a stride of two are inserted to compress the speech encoder output by a factor of eight. The model's decoder is initialized from mBART decoder and is shared by two data modalities. We alter the original model architecture to decoupled the mBART output layer and embedding layer instead of using a shared projection layer.

2.1 Pretrained models

We use two state-of-the-art pretrained models — Wav2Vec 2.0 and mBART — for speech and text data, respectively. Both models were trained independently with self-supervised tasks and then finetuned with the corresponding ASR and MT tasks using labeled data.

Wav2Vec 2.0 Wav2Vec 2.0 is a powerful transformer based framework pretrained on self-supervised tasks with large amount of unlabeled speech data (Baevski et al., 2020). There are three main modules in Wav2Vec 2.0 model. The feature encoder is a convolution neural network, which takes wave-form audio as inputs and converts them into a sequence of continuous feature vectors. Then the quantization module learns the latent discrete speech features from the continuous embeddings by sampling from Gumbel softmax distribution (Jang et al., 2017) using two codebooks of size 320. Finally, a transformer based context encoder extracts high quality contextual speech representations from the features. By finetuning on speech data with transcriptions, Wav2Vec 2.0 achieves outstanding performance on ASR task.

In this work, we adopt the Wav2Vec large model finetuned for ASR task ("wav2vec-vox-960h-pl") (Xu et al., 2020). The context encoder in the model has 24 transformer layers with 16 attention heads, and the hidden dimension is 1024. The model was pretrained on Librispeech and LibriVox audio corpus and then finetuned on 960 hours of transcribed Librispeech data (Panayotov et al., 2015), Libri-light data (Kahn et al., 2020a), and pseudo-labeled audio data (Kahn et al., 2020b).

mBART mBART is a sequence-to-sequence encoder-decoder architecture pretrained on large-scale multilingual unlabeled text corpus (Liu et al., 2020). During pretraining, mBART is trained as a denoising auto-encoder which reconstructs the corrupted input text to its original form. The pretrained mBART was fintuned with paralleled machine translation data and achieved significant performance gains on multilingual machine translation (MT) task. For this work, we used the mBART-large-50-one-to-many model, which consists of a 12-layer transformer encoder and a 12-layer transformer decoder. The model was pretrained on 50 languages and finetuned to translate English to the other 49 languages (Tang et al., 2020).

2.2 Multimodal training objectives

During training, both S2T translation and T2T translation tasks are performed using an online knowledge distillation process that mitigates the speech-text modality gap with the following loss function:

$$l = l_{st} + l_{t\_guide} + l_{mt} + l_{cross\_attn} \quad (1)$$

where l_{st} and l_{mt} are cross entropy loss between ground truth and hypothesis from speech and text inputs respectively, l_{t_guide} is the cross entropy loss between hypothesis from speech and text, and l_{cross_attn} is the cross attention regularization from two input data modalities (Tang et al., 2021b).

2.2.1 Dynamic Dual Skew Divergence

To improve the text-guided learning in joint task training, we replace the cross-entropy based text guide loss from eq. 1 with a loss based on Kullback-Leibler divergence that considers S2T translation errors from (1) generating an unlikely hypothesis and (2) not generating a plausible hypothesis when compared with the T2T translation. In previous studies, similar approaches have shown promising results when applied to machine translation task (Li et al., 2021b) and measuring text generation performance (Pillutla et al., 2021).

Kullback-Leibler Divergence Kullback-Leibler (KL) divergence measures the divergence of probability distributions $S(x)$ from $T(x)$:

$$D(T||S) = \sum T(x) \log \frac{T(x)}{S(x)} \quad (2)$$

We denote $T(x)$ as the translation hypothesis probability distribution from the text input and $S(x)$ as the probability distribution from the speech input. $D(T(x)||S(x))$ is an asymmetric distance metric that measures the deviation of S2T distribution with the T2T distribution (type II error). If we switch the sides of $T(x)$ and $S(x)$, minimizing $D(S(x)||T(x))$ emphasizes errors caused by hypotheses generated from the S2T task that are not

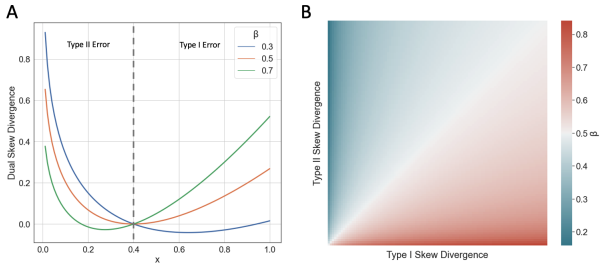

Figure 1: A) Depending on the dominant error types, higher or lower value of β tilts the dual skew divergence curve and providing a steeper slope of the loss curve for current training state. X axis represents S2T output, T2T output is set to 0.4 in this example. B) Value of β dynamically changes based on the values of type I and type II skew divergence

likely to be generated from the T2T task (type I error).

Dual Skew Divergence The definition of KL divergence holds when the observed distribution (e.g. $S(x)$ in the case of $D(T||S)$) is non-zero. However, during training, the probabilities of some tokens can go towards zero due to the large vocabulary size of mBART. To mitigate this issue, in dual divergence, we replace the KL divergence with the skew divergence:

$$D_s(T||S) = D(T||\alpha T + (1-\alpha)S) \quad (3)$$

where α is a hyperparameter. In this study, we set α to 0.01 for all experiments.

To mitigate the modality gap between speech and text inputs, we consider both types of errors with dual skew KL divergence in training:

$$D_{ds}(T,S) = \beta D_s(S||T) + (1-\beta)D_s(T||S) \quad (4)$$

where β is a weight to balance the two types of errors. When using dual skew divergence as a loss function during training, the value of β affects convergence depending on the dominant error type at the current step. When S2T task undergenerates the probability distribution output by T2T task (higher type II error), a lower value of β motivates faster learning with higher magnitude of gradient. While type I error dominates, a higher value of β is favored by training instead (Figure 1A).

Dynamic Weight As the dominant error type could change during training, we dynamically tune the value of β in eq. 4 based on the values of two dual skew divergence components at each training step. We first normalize the skew divergence to achieve a value bounded between 0 and 1.

$$M(S||T,\beta) = \frac{\log(1+\beta D_s(S||T))}{(1+\log(1+\beta D_s(S||T)))} \quad (5)$$

And then we solve for the value of β that maximizes the product of two measures derived with above equation:

$$\beta = \arg\max \Big((M(S||T,\beta) * M(T||S,1-\beta) \Big) \quad (6)$$

This logic ensures that β is constantly updated based on type I and type II skew divergence to achieve the preferred dual skew divergence for the current training step (Figure 1B).

3 Finetuning Approaches

To avoid overfitting and moderate generalization, we finetune the base model with a proposed sampling loss algorithm. In addition, we experiment with the effect of finetuning on languages with similar linguistic typology or vocabulary to see if there is negative transferring with the multilingual setting. Finally, we test the consequence of using in-domain data.

The motivation for sampling loss comes from a hypothesis that the ground truth translations may lack diversity. We can make the translation model more robust and increase end-phrase diversity by training with alternate translations to supplement the ground truth translations. To achieve this, we clone the T2T components from the trained base model and use beam search as a mechanism to generate the alternate translations to guide the S2T components. During the beam search, the target probabilities of all the nodes visited are considered during loss computation as illustrated in Figure 2. We reuse the dynamic dual skew divergence loss to train the student model, and this is the only loss applied during our sampling loss finetuning. We recognize that other sampling strategies could also generate alternative translations.

A similar approach is explored in mixed cross entropy loss(Li and Lu, 2021). While mixed cross entropy loss achieves the same effect as sampling loss, sampling loss considers the complete target distribution as ground truth while training the student model.

```
Ground Truth: Today is a wonderful day.

Top 3 beams:
Today is a good  day.
This  is a good  day.
Today is a great day.

At timestep 5, DDSD uses target distribution
P(y | X, Today is a wonderful)

At timestep 5, Sampling Loss uses target distributions
P(y | X, Today is a good)
P(y | X, This  is a good)
P(y | X, Today is a great)
```

Figure 2: Sampling loss example with beam width=3. All target distributions are considered for loss computation.

3.1 Sampling Loss

3.2 Language Grouping

Several studies (Prasanna, 2018; Sachan and Neubig, 2018; Tan et al., 2019; Fan et al., 2021) have suggested that multilingual MT models benefit from training models with languages sharing similar linguistic features. In this work, we experiment with two grouping strategies. One is based on linguistic typology where German and Chinese are considered as subject–verb–object (SVO) languages[1] while Japanese is a subject–object–verb (SOV) language. The other is based on vocabulary sharing. Japanese kanji was derived from Chinese characters, and most of the time the meaning are the same or very similar. For this reason, we consider Japanese and Chinese as a shared-vocabulary group.

3.3 Domain Adaption

Finetuning is a popular approach for domain adaption in MT to boost model performance (Freitag and Al-Onaizan, 2016; Luong and Manning, 2015). As the IWSLT 2022 task uses TED talks as the test data, we evaluate the effect of finetuning our base model using the MuST-C V2 (Di Gangi et al., 2019) dataset, a multilingual speech translation corpus comprising English audio recordings from TED talks.

4 Experimental Setup

In this section, we first describe the datasets and hyperparameters settings used in our model training experiments, followed by a brief introduction of our audio segmentation approach that improves our model performance on unsegmented datasets.

[1]There is a small part of German is SOV.

4.1 Data

We train our models using MuST-C V2 (Di Gangi et al., 2019), CoVoST v2 (Wang et al., 2020) and Europarl-ST V1.1 train-clean dataset (Iranzo-Sánchez et al., 2020). The entire corpus contains paired audio-text samples for Speech Translation, including transcriptions of the source audios. MuST-C supports en-to-14 languages, including en→de, en→ja and en→zh. CoVoST supports en-to-15 languages, again including en→de, en→ja and en→zh. However, as Europarl-ST provides translation data between six European languages, only en→de is supported. Table 1 presents statistics on the datasets. We discard short audio clips of less than 50ms and long audio clips of greater than 30s. We hold out 1% of the data as the development set. Additionally, we evaluate our models using the unsegmented test set released for IWSLT 2019 and IWSLT 2020.

4.2 Training Details

We use the fairseq[2] library to train our models. For the base model using the cross-entropy as the text-guided loss, we set the loss weights of l_{st}, l_{t_guide}, l_{mt}, and l_{cross_attn} as 0.2, 0.8, 1.0, and 0.02, respectively. When training using the DDSD text-guided loss, we reduce l_{mt} to 0.2. For the finetuning experiments, the beam size is set to 1 for the sampling loss algorithm. We set dropout to 0.3. We use the Adam optimizer (Kingma and Ba, 2017) and inverse square root scheduler with an initial learning rate of 1e-8. We set the warm-up phase to 5000 steps and the training batch size to a maximum of three for both the base and finetuned models. The model parameters are updated every four batches; the maximum number of iterations is set to 120,000 for the base models, while we train the finetuned models until convergence with the early stopping strategy when the loss on the validation set increases for three consecutive evaluations. Each model is trained on eight NVIDIA V100 GPUs for around 24 to 48 hours.

4.3 Speech Segmentation

Previous years' IWSLT results show that the segmentation approach has significant impact on the performance of end-to-end speech translation (Ansari et al., 2020; Anastasopoulos et al., 2021). We use the WebRTCVAD[3] toolkit to split the unseg-

[2]https://github.com/pytorch/fairseq
[3]https://pypi.org/project/webrtcvad

	MuST-C			CoVoST			Europarl-ST
	en → de	en → ja	en → zh	en → de	en → ja	en → zh	en → de
Samples (in thousands)	238.0	314.0	343.9	289.0	289.0	289.0	31.3
Average audio length (s)	6.3	5.8	5.8	5.4	5.4	5.4	8.5
Average source text length (tokens)	27.2	25.5	25.5	17.8	17.8	17.8	31.4
Average target text length (tokens)	27.9	24.2	22.9	18.3	19.2	15.7	36.5

Table 1: Dataset statistics

Stage	Length threshold (s)	WebRTCVAD		
		A	FD(ms)	ST
1	0	1	10	0.9
2	21	3	30	0.9
3	30	3	10	0.5
4	21	-	-	-

Table 2: Parameter used at each stage of speech segmentation. We pick 21 seconds and 30 seconds as length thresholds as they represent the 99.5% percentile and max of the audio length of our training data. (A: aggressiveness, FD: frame duration, ST: silence threshold)

	Seg. Stage	BLEU	#Seg.	Seg. Length (P25/P50/P75)
IWSLT 2019	S1	23.21	2384	2.29/4.12/7.90
	S2	23.27	2881	2.32/4.06/7.43
	S3	23.27	2909	2.31/4.05/7.41
	S4	**25.00**	963	14.96/17.80/19.58
IWSLT 2020	S1	23.61	2071	2.40/4.21/7.74
	S2	24.42	2408	2.38/4.19/7.22
	S3	24.38	2464	2.37/4.16/7.22
	S4	**26.58**	811	15.07/17.78/19.68

Table 3: Speech translation performance on unsegmented development sets at each segmentation stage. All results are based on the DDSD_{de} model.

mented audio data with a multi-stage segmentation and merging strategy. In each of the first three stages, we split audios that are longer than a corresponding threshold with gradually increased aggressiveness. In the last stage, we merge the short audios from left to right until the merged audio reaches a certain length Table 2. This strategy generates audio segments that are neither too long to be processed by the end-to-end speech translation model nor too short to convey enough contextual information. Throughout this paper we refer to this as our 'own' segmentation.

5 Results and Analyses

In this section, we present our experimental results and analyses. All the reported results are obtained from a single run using one of the following model settings:

- **CE**: This is our baseline model which uses cross-entropy as the text-guided loss .
- **DDSD**: This model uses the DDSD described in Section 2.2.1 as the text-guided loss.
- **DDSD+DDSD**: This is a finetuned model where both of the base and finetuning training are using the DDSD as the text-guided loss.
- **DDSD+SL**: This is a finetuned model where the text-guided loss of the base and the finetuning training are the DDSD and the sampling loss algorithm explained in Section 3.1, respectively.

The corpora and target languages used in a model training are denoted in superscript and subscript, respectively. If no superscript or subscript appears, all the available corresponding corpora or target languages have been used. For example, DDSD_{de} means a bilingual en→de model trained using all the corpora mentioned in Section 4.1.

As for the evaluation datasets, if our model can directly handle the size of a given audio clip, such as the audio in the MuST-C dataset, we directly use the provided data. Otherwise, we use the algorithm described in Section 5.1 to split audio clips into smaller chunks.

5.1 Effect of Speech Segmentation

We tune the speech segmentation algorithm described in Section 5.1 using the IWSLT 2019 and IWSLT 2020 development sets. Table 3 summarizes the performance of the DDSD_{de} model at each segmentation stage. Since few segments have audio lengths longer than 30 seconds, Stage 3 only results in a minimal change to the number of segments and the audio length distribution. After merging short audio clips in Stage 4, the model performance improves by +1.73 and +2.20 BLEU points for the IWSLT 2019 set and IWSLT 2020 set respectively. We hypothesize that this improvement is the result of the model's ability to access more contextual information, and therefore generate better translations. For the rest of the experiments, we report

Model	IWSLT 2019* en → de	IWSLT 2020* en → de	Must-C COMMON en → de	en → ja	en → zh
CE_{de}	23.98	26.02	29.71	-	-
$DDSD_{de}$	**25.00 (+1.02)**	**26.58 (+0.56)**	**30.59 (+0.88)**	-	-
CE	23.25	24.44	28.46	16.27	25.41
DDSD	**24.20 (+0.95)**	**25.67 (+1.23)**	**30.25 (+1.79)**	**16.77 (+0.5)**	**26.69 (+1.28)**

Table 4: Comparison of results using cross-entropy (CE) and the DDSD text-guided loss. Numbers in parentheses show the BLEU difference between models using DDSD and CR losses. * indicates own segmentation.

finetuning Approach	Model	IWSLT 2019* en → de	IWSLT 2020* en → de	Must-C COMMON en → de	en → ja	en → zh
Sampling Loss	$DDSD_{de}$+SL_{de}	**+0.13**	**+0.33**	-0.43	-	-
	DDSD+SL	**+0.07**	**+0.02**	-0.07	**+0.13**	**+0.03**
Language Grouping: Linguistic Typology	DDSD+$DDSD_{de,zh}$	-0.15	-0.03	**+0.13**	-	**+0.02**
	DDSD+$DDSD_{ja}$	-	-	-	**+0.3**	-
Language Grouping: Vocabulary Sharing	DDSD+$DDSD_{ja,zh}$	-	-	-	**+0.44**	-0.10
	DDSD+$DDSD_{de}$	**+0.22**	**+0.17**	**+0.3**	-	-
Sampling Loss + Vocabulary Sharing	DDSD+$DDSD_{ja,zh}$+$SL_{ja,zh}$	-	-	-	**+0.48**	**+0.02**
	DDSD+$DDSD_{de}$+SL_{de}	-0.03	**+0.34**	**+0.36**	-	-
Domain Adaption	DDSD+$DDSD^{Must\text{-}C}$	**+0.08**	**+0.25**	+0.00	**+0.27**	-0.03

Table 5: Relative results of using different finetuning approaches compared with their base model, where numbers in bold mean the finetuned model has a higher BLEU score compared with its base model. * indicates own segmentation

results using segments generated at Stage 4 for the IWSLT 2019 and IWSLT 2020 development sets.

5.2 Effect of the DDSD

We train en→de translation models as well as one-to-many multilingual models using the cross-entropy loss or the DDSD loss as the text-guided loss, with the evaluation results presented in Table 4. From our experiments, en→de models always outperforms the multilingual models. However, the DDSD loss effectively reduces the quality gap between the bilingual and multilingual models from an average of -1.19 BLEU to -0.68 BLEU. Models with DDSD loss consistently outperform those with cross-entropy text-guided loss on all the tested language arcs for both en→de and multilingual models. The BLEU score improvement is in the range of +0.5 to +1.8, where the smallest +0.5 BLEU improvement is observed for the multilingual model's en→ja arc.

5.3 Effect of finetuning

We study three types of finetuning modifications: using the sampling loss, finetuning with language-based groupings and domain adaptation. Since DDSD has consistently improved BLEU metric values, all of our finetuning experiments use models initialized from those trained with the DDSD text-guided loss in the previous section. Table 5 summarizes the change in BLEU score of the proposed approaches comparing to the respective base model trained with DDSD text-guided loss.

Sampling Loss We experiment with the proposed sampling loss algorithm from Section 3.1 and report the results at the first two rows of Table 5. We observe mixed results when comparing $DDSD_{de}$ and DDSD models in Table 4. One explanation is that the base model has been trained with enough data diversity, and therefore the sampling loss has limited influence.

Language Grouping For the linguistic-typology-based finetuning, the finetuned DDSD+$DDSD_{de,zh}$ model (SVO languages) behaves almost the same as the base DDSD model. On the other hand, the vocabulary-sharing-based finetuned model, DDSD+$DDSD_{ja,\,zh}$, achieves a moderate +0.44 BLEU improvement on the en→ja arc while having a small degradation of -0.10 BLEU on the en→zh arc. These results suggest that the en→zh arc which is included in both of the language groups is not affected by either of the language grouping strategies. However, it is worthy to note that the result of en→ja finetuning (+0.3 BLEU) falls behind the en→ja+zh multilingual finetuning (+0.48 BLEU). We also consider finetuning the vocabulary-sharing-based models using the sampling loss where we don't observe consistent improvements in this set of results.

Model	Test set	Language	Segmentation	BLEU ref2	BLEU ref1	BLEU both
$DDSD_{de}+SL_{de}$	IWSLT 2022	en→de	own	22.6	20.1	31.5
	IWSLT 2021	en→de	own	24.4	20.6	34.5
			given	21.9	17.9	30.1
$DDSD+DDSD_{ja,zh}+SL_{ja,zh}$	IWSLT 2022	en→ja	own	15.3	16.2	25.3
		en→zh	own	30.4	30.8	37.9

Table 6: Performance of the submitted systems on IWSLT 2022 test sets and progression test set.

Domain Adaption We finetune the base model only using the Must-C dataset and report the results in the last row of the Table 5. Apart from increases of +0.27 and +0.25 BLEU score on the en→ja Must-C testset and en→de IWSLT 2020 testset respectively, there is little-to-no effect on the other testsets. One possible explanation is that the base model has been trained using a fair amount of the representative data, and therefore, the model cannot benefit from further finetuning on the Must-C dataset.

5.4 Submission

Based on the results obtained from the IWSLT development datasets and Must-C COMMON test sets, we submitted $DDSD_{de}+SL_{de}$ and $DDSD+DDSD_{ja,zh}+SL_{ja,zh}$ as our primary systems for en→de and en→ja+zh with our own segmentation.

We present the results on the IWSLT 2022 and IWSLT 2021 test sets in Table 6. Our systems achieved 22.6, 15.3, and 30.4 BLEU scores on the IWSLT 2022 en→de, en→ja and en→zh blind test sets, respectively. On the en→de progression test set (IWSLT 2021), our system scored 24.4 with our own segmentation and 21.9 with the provided segmentation. Note that the IWSLT 2021 best BLEU scores on same test sets were 24.6 and 21.8 for own segmentation and provided segmentation, respectively, and both results were from cascaded systems (Anastasopoulos et al., 2021).

6 Conclusion

In this paper, we adapt and improve the existing dual skew divergence loss by dynamically balancing the model's quality and diversity via the DDSD text-guided loss. The DDSD text-guided loss outperforms the baseline cross-entropy loss on all the experimented language arcs. We observe that for CE and DDSD loss, one-to-one models always outperform one-to-many multilingual models, however DDSD reduces the performance gap between them. We also consider three different finetuning approaches: sampling loss, language grouping, and domain adaption. Overall, mixed results are observed and none of the finetuning strategies stand out from the others. In addition, the results of the segmentation experiments indicate that the translation quality can be boosted by presenting audios that are longer than the majority of the training data since more context can be taken into consideration. Our submitted end-to-end speech translation system achieves on par performance with the best cascaded system from IWSLT 2021.

References

Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha Elbayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changhan Wang, and Matthew Wiesner. 2021. FINDINGS OF THE IWSLT 2021 EVALUATION CAMPAIGN. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 1–29, Bangkok, Thailand (online). Association for Computational Linguistics.

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondřej 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, Alexander Waibel, and Changhan Wang. 2020. FINDINGS OF THE IWSLT 2020 EVALUATION CAMPAIGN. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 1–34, Online. Association for Computational Linguistics.

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

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.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast domain adaptation for neural machine translation.

J. Iranzo-Sánchez, J. A. Silvestre-Cerdà, J. Jorge, N. Roselló, A. Giménez, A. Sanchis, J. Civera, and A. Juan. 2020. Europarl-st: A multilingual corpus for speech translation of parliamentary debates. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8229–8233.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax.

J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, Q. Xu, P.E. Mazare, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. 2020a. Libri-light: A benchmark for asr with limited or no supervision.

Jacob Kahn, Ann Lee, and Awni Hannun. 2020b. Self-training for end-to-end speech recognition. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Haoran Li and Wei Lu. 2021. Mixed cross entropy loss for neural machine translation. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 6425–6436. PMLR.

Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2021a. Multilingual speech translation with efficient finetuning of pretrained models.

Zuchao Li, Hai Zhao, Yingting Wu, Fengshun Xiao, and Shu Jiang. 2021b. Controllable dual skew divergence loss for neural machine translation.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation.

Minh-Thang Luong and Christopher Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign*, pages 76–79, Da Nang, Vietnam.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers.

Raj Noel Dabre Prasanna. 2018. Exploiting multilingualism and transfer learning for low resource machine translation.

Devendra Singh Sachan and Graham Neubig. 2018. Parameter sharing methods for multilingual self-attentional translation models. *arXiv preprint arXiv:1809.00252*.

Xu Tan, Jiale Chen, Di He, Yingce Xia, Tao Qin, and Tie-Yan Liu. 2019. Multilingual neural machine translation with language clustering. *arXiv preprint arXiv:1908.09324*.

Yun Tang, Hongyu Gong, Xian Li, Changhan Wang, Juan Pino, Holger Schwenk, and Naman Goyal. 2021a. Fst: the fair speech translation system for the iwslt21 multilingual shared task.

Yun Tang, Juan Pino, Xian Li, Changhan Wang, and Dmitriy Genzel. 2021b. Improving speech translation by understanding and learning from the auxiliary text translation task.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning.

Changhan Wang, Anne Wu, and Juan Pino. 2020. Covost 2: A massively multilingual speech-to-text translation corpus.

Qiantong Xu, Alexei Baevski, Tatiana Likhomanenko, Paden Tomasello, Alexis Conneau, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. 2020. Self-training and pre-training are complementary for speech recognition.