Towards Unsupervised Speech-to-Speech Translation

Anonymous ACL submission

Abstract

Most of the speech-to-speech translation models heavily rely on parallel data, which is hard to collect especially for low-resource languages. To tackle this issue, we propose to build a speech-to-speech translation system without leveraging any kind of paired data. To the best of our knowledge, this work is the first one that has successfully built a speech-to-speech translation system under an unsupervised scenario. We use fully unpaired data to train our unsupervised system and make comparable results with the other supervised methods proposed just a few years ago. Furthermore, to demonstrate that our method can generalize well across different languages, we evaluate our system on CVSS, a multi-lingual speech-to-speech corpus, and get promising results in different translation directions.

1 Introduction

Speech-to-speech translation (S2ST) converts speech from one language to another, attempting to bridge the communication barriers between people speaking different languages. Conventionally, an S2ST system is accomplished by concatenating the three components: automatic speech recognition (ASR), text-to-text machine translation (MT), and text-to-speech (TTS) synthesis sub-modules (Lavie et al., 1997; Wahlster, 2000; Nakamura et al., 2006). Recently, with works on direct or end-to-end speech-to-text translation (ST; Bérard et al., 2018; Inaguma et al., 2019; Gangi et al., 2019), conducting 2-cascade systems (ST→TTS) may also be an option for S2ST. Moreover, studies on S2ST without leveraging intermediate text representations are emerging, such as direct S2ST (Jia et al., 2019, 2021) and cascade S2ST based on discrete units or representation (Tjandra et al., 2019; Lee et al., 2021).

However, most S2ST and ST systems are trained under supervision, making them heavily rely on parallel data. For cascade systems, each component requires its corresponding labeled data. Although it can be collected separately, using data from different corpora might introduce domain mismatches simultaneously. On the other hand, direct S2ST and ST systems are more constrained by the limited amount of labeled data. Unlike ASR or MT, parallel data for ST and S2ST is scarce and even harder to collect, especially for low-resource languages.

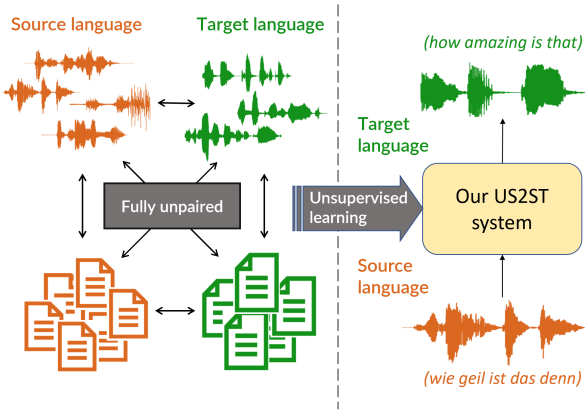

Figure 1: Overview of our unsupervised speech-to-speech translation (US2ST) system.

In contrast to parallel data, unlabeled data is much easier to obtain regardless of modalities. Even though the unlabeled data is broadly accessible, learning without supervision, namely unsupervised learning, can be extraordinarily challenging. Thus, previous works on unsupervised MT (UMT) started from word-level translation. For instance, Cao et al. (2016); Zhang et al. (2017); Conneau et al. (2017) perform UMT by learning cross-lingual alignments of word embeddings. These alignment-based methods have demonstrated the possibilities of unsupervised learning in translation-related tasks. Meanwhile, Artetxe et al. (2017); Lample et al. (2017) further improve the performance of UMT by introducing the idea of back-translation (Sennrich et al., 2015a). Nowadays,

with cross-lingual pre-trained models, the results of UMT are even comparable with some previous supervised methods (Lample and Conneau, 2019; Liu et al., 2020)

Despite the breaking-through of UMT, unsupervised ST (UST; Chung et al., 2018, 2019b) and unsupervised S2ST (US2ST) remain rarely studied. Motivated by recent success in unsupervised ASR (UASR; Baevski et al., 2021; Liu et al., 2022a) and unsupervised TTS (UTTS; Ni et al., 2022; Liu et al., 2022b), in this work, we proposed a cascade US2ST framework, dealing with both ST and S2ST tasks. To our best knowledge, this is the first paper that tackles S2ST under a completely unsupervised scenario.

We evaluate our cascade US2ST system on CVSS (Jia et al., 2022), a multi-lingual S2ST corpus. The corpus is built on top of an ST corpus, CoVoST 2 (Wang et al., 2020b), thus we also evaluate our UST performance on the ST corpus. We demonstrate that our system can generalize well across different languages by conducting experiments on three different translation directions. Despite that our system learns with only unlabeled data, the results are still promising. On CoVoST 2, our system not only shows comparable results with some supervised direct ST but also outperforms them in some translation directions. For S2ST, our system can generate natural and clean speech, with less than 5.0 BLEU score degradation compared to our supervised upper bound. To demonstrate that our results might be as good as some supervised S2ST models, the audio samples are also available on the website[1].

2 Related works

S2ST Typically, conventional S2ST systems are composed of ASR, MT and TTS subsystems (Lavie et al., 1997; Wahlster, 2000; Nakamura et al., 2006). With research on direct ST (Bérard et al., 2016; Weiss et al., 2017; Li et al., 2020), conducting 2-cascade S2ST systems (ST→TTS) is also available and can alleviate the error propagation issue between ASR and MT. Besides the cascade systems, recent research also focus on direct S2ST. Jia et al. (2019) proposed Translatotron, the first sequence-to-sequence S2ST model that can be directly trained end-to-end on multi-objectives; and the follow-up work, Translatotron 2 (Jia et al., 2021), further bridged the performance gap between direct S2ST and ST→TTS cascade systems.

Apart from direct S2ST, some works intend to learn S2ST models with discrete units or representations instead of intermediate text representations, aiming to solve S2ST for unwritten or untranscribed languages. Tjandra et al. (2019); Zhang et al. (2021) proposed to learn their S2ST models through discrete tokens generated by vector quantized variational autoencoder (VQ-VAE) related techniques. Lee et al. (2021) proposed to utilize discrete units from HuBERT (Hsu et al., 2021) and showed comparable results with some text-based methods, indicating their potential on S2ST between unwritten languages.

Among all the works regarding S2ST, none of them is under unsupervised scenarios. Even for unwritten S2ST, speech-to-speech data from the source language to the target language is required.

UMT and UST Different from US2ST, UMT has been studied for a long time, trying to solve the translation problem with mono-lingual data only. Starting from word-level translation, Cao et al. (2016) learns bilingual word embeddings by distribution matching; while (Zhang et al., 2017; Conneau et al., 2017) focuses on learning the cross-lingual mapping between the monolingual word embeddings through adversarial training. These methods can only perform word-by-word translation, which might lead to unnatural sentences. Based on previous work on cross-lingual word embeddings, Artetxe et al. (2017); Lample et al. (2017) further improve the performance of UMT by introducing the idea of back-translation (Sennrich et al., 2015a) and denoising auto-encoder (DAE; Vincent et al., 2008). Moreover, Lample et al. (2018) demonstrated that UMT can be accomplished with suitable initialization of the translation models, language modeling and iterative back-translation.

Over the past few years, language model pre-training has brought a significant impact on natural language understanding (NLU) and cross-lingual understanding (XLU). Lample and Conneau (2019) proposed two methods to learn cross-lingual language models (XLM), showing that cross-lingual pre-training may also benefit UMT. Their results of UMT outperformed previous SOTA and were even comparable with some supervised models.

Compared with UMT, UST remains rarely studied. Chung et al. (2018) proposed a method based on cross-modal alignments between spoken word

[1] https://acl2022anonymous.github.io/us2s-demo

and text word embeddings, performed word-level UST. Furthermore, to improve the quality of translation, they integrated the method with a pre-trained language model (LM) and DAE (Chung et al., 2019b).

UASR and UTTS UASR takes audio features or representations as input and generates phoneme sequences without supervision. To tackle the challenging problem, Liu et al. (2018) first came out with the idea of applying a Generative Adversarial Network (GAN) (Goodfellow et al., 2020). However, phoneme-level boundaries are required to segment the audio and construct embedding sequences. (Chen et al., 2019) breaks the limit by iteratively refining the audio segments with Hidden Markov Model (HMM) and GAN, achieving completely UASR.

Recently, Baevski et al. (2021) proposed wav2vec-U, building the GAN-based UASR framework on top of the representation from wav2vec 2.0 (W2V2) (Baevski et al., 2020), a self-supervised speech model. The results outperformed previous SOTA, and are even comparable with some of the best-known supervised methods. Moreover, the original paper has shown that with the cross-lingual pre-trained version of W2V2 (Conneau et al., 2020), UASR in other languages is also available. The follow-up work, wav2vec-U 2.0 (Liu et al., 2022a), enabled the model to be trained end-to-end with the simplified pipeline and the improved training objective.

Inspired by the recent success of UASR, Ni et al. (2022); Liu et al. (2022b) accomplished UTTS by leveraging pseudo labels from wav2vec-U or wav2vec-U 2.0. The results of their UTTS models achieve comparable performance against those trained on the true labels.

3 Methods

Our cascade US2ST architecture is composed of three components: UASR, UMT, and UTTS, as indicated in Fig. 2. We trained each of the component separately with fully unpaired data. During inference, we concatenate all of them and form the functionality of S2ST. In this section, We will go through the details of each sub-module individually from sections 3.1 to 3.3.

3.1 UASR

We conduct UASR subsystem following wav2vec-U (Baevski et al., 2021), and our code is based on their implementation in fairseq[2]. Besides its breakthrough performance on UASR in multiple languages, the robustness and stabilities across different corpora have also been analyzed (Lin et al., 2022).

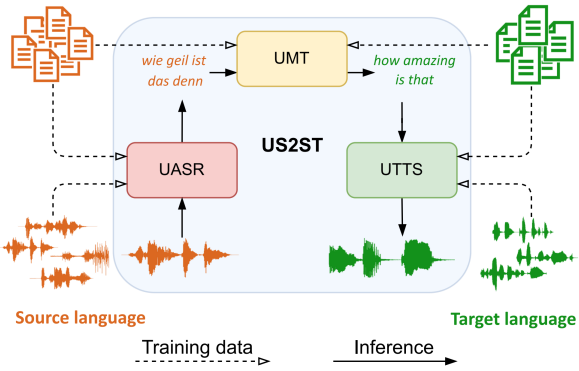

Figure 2: The framework of our cascade US2ST.

Wav2vec 2.0 Wav2vec-U utilizes the inner-layer representations from wav2vec 2.0 (W2V2). Recent studies have shown that the layerwise representations of the transformer model might be informative (Pasad et al., 2021) and can be adopted for different downstream tasks with promising results (Yang et al., 2021). In this work, following wav2vec-U, we also leverage the intermediate representations extracted from the transformer of the pre-trained W2V2.

Audio preparation First, we detect the silence segments in the raw audio by rVAD, which is an unsupervised voice detection method (Tan et al., 2020). We then remove all of the detected silence segments, making the audio more compact. Next, we feed the compact audio into the pre-trained W2V2, and extract single-layer representations from it. After getting the W2V2 representations, we follow the dimension reduction procedures proposed in wav2vec-U to get more compact sequences for later training.

Text preparation We normalize our text data by removing the punctuation marks and making them lowercase. Then, we phonemicize each sentence using an off-the-shelf Phonemizer[3] (Bernard and Titeux, 2021), which supports the phonemicization across multiple languages based on the International Phonetic Alphabet (IPA). Since the silence removal process adopted in audio prepara-

[2] https://github.com/facebookresearch/fairseq
[3] https://github.com/bootphon/phonemizer

tion might be incomplete, wav2vec-U tackle such a problem by introducing a silence token (<SIL>) into the set of generator outputs. For better simulation of the generator output behavior, we insert <SIL> at the beginning and at the end of each phoneme sequence. Moreover, random <SIL> insertion between word boundaries at a certain rate is also adopted, suggested by wav2vec-U.

GAN-based training Generative adversarial nets (GAN; Goodfellow et al., 2020) are composed of a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$. The generator $\mathcal{G}$ aims at producing samples that are undistinguished by $\mathcal{D}$; while the discriminator $\mathcal{D}$ learns to classify if the samples come from training data or $\mathcal{G}$. In wav2vec-U, both $\mathcal{G}$ and $\mathcal{D}$ are simple convolutional neural networks (CNN). The training objective of wav2vec-U consists of the original GAN objective with gradient penalty (Gulrajani et al., 2017), a segment smoothness penalty, and a phoneme diversity penalty.

Decoding & self-training Once the UASR models were trained, we integrate the generator outputs with some decoding strategies to get the final text sequences. Besides traditional Viterbi decoding, lexicon-based kenlm decoder[4] and the weighted finite-state transducer (Mohri et al., 2002) are also introduced as different decoding methods. To make further improvements, we also apply self-training on Hidden Markov Models (HMM; Rabiner and Juang, 1986). The procedure of self-training is described as follows: First, we use the decoded outputs from the generator as pseudo labels; next, we train the HMM through these pseudo labels along with the speech representations extracted from the W2V2; finally, we obtain word-level or phoneme-level sequences with WFST decoding. Note that the representations come directly from pre-trained W2V2 without modifications.

3.2 UMT

We implement the UMT by following XLM (Lample and Conneau, 2019), which initializes a seq2seq model by pre-trained XLM, and train the model with online back-translation.

XLM XLM is a cross-lingual transformer language model, which pre-trains on sentences from two or more languages. The shared Byte Pair Encoding (BPE; Sennrich et al., 2015b) vocabularies and masked-based language model pre-training provide embeddings with multilingual alignment and contextualized information (Lample et al., 2018; Lample and Conneau, 2019).

Masked language modeling (MLM) The objective of XLM pre-training is masked language model (MLM) loss. We sample some of the tokens from a sentence randomly and substitute the sampled token with (1) a special token [MASK] (2) random tokens (3) left unchanged. The model aims to recover the masked tokens from the noisy sentences. Moreover, to prevent the model from predicting common words, we calculate the frequencies of the tokens in sentences, and the sample weight is proportional to its inverted frequency. The Masked language modeling captures the relation between tokens, which learns contextualized understanding.

Back-translation Let $u^*(y)$ be the transcribed sentence of y, which belongs to the source language $\mathcal{S}$; $v^*(x)$ be the transcribed sentence of x, which belongs to the target language $\mathcal{T}$. The $(u^*(y), y)$ and $(v^*(x), x)$ form the pseudo-parallel data, and the objective for the translation model is to map $u^*(y)$ and $v^*(x)$ back to y and x respectively, the back-translation loss is as follows:

$$\begin{aligned}\mathcal{L}_{back} =& \mathbb{E}_{x\in\mathcal{T}}[-\log P_{s\to t}(y|u^*(y))] \\ &+\mathbb{E}_{y\in\mathcal{S}}[-\log P_{t\to s}(x|v^*(x))]\end{aligned} \quad (1)$$

In the first few steps, we add denoising auto-encoder loss to help the training process, and the weight of the denoising auto-encoder will decay as the iteration increases.

3.3 UTTS

Some works intend to improve the performance of TTS through unlabeled data. For instance, pre-training the encoder/decoder (Chung et al., 2019a); utilizing the dual nature of TTS and ASR tasks (Ren et al., 2019); applying variational auto-encoder to learn from speech disentanglement (Lian et al., 2022).

In spite of the improvement they brought, these methods still depend on certain levels of paired data. Directly training a UTTS without any supervision from paired data seems to be extremely hard. However, with recent success in UASR, UTTS might be accomplished in another way—training on the pseudo labels generated from UASR systems (Ni et al., 2022; Liu et al., 2022b).

[4] We adopt the implementation in fairseq which has integrated the functionalities from flashlight

Table 1: The training data for each part of the US2ST system and the corresponding evaluation corpus. Wiki stands for Wikipedia; CV4 is the abbreviation for Common Voice version 4. The $*$ indicates mono-lingual data.

	Text	Audio
UASR	Wiki	CV4
UMT	WMT'14 + Wiki	-
UTTS	LibriSpeech LM*[6]	LJspeech*

4 Experiments

4.1 Data

To demonstrate our US2ST systems across different languages, we evaluate our results on CVSS (Jia et al., 2022), which is a newly proposed multilingual S2ST corpus based on CoVoST 2 (Wang et al., 2020b) and Common Voice ver.4 (CV4; Ardila et al., 2019). By the transcriptions files from CV4 and translation data from CoVoST 2, we are able to evaluate each of our components individually.

However, we do not utilize any paired data from the corpus during training stage; instead, we use audio and text data from different corpora, constructing a fully unpaired scenario for our US2ST. For audio, we adopt Common Voice ver.4 for UASR and LJspeech (Ito and Johnson, 2017) for UTTS without using any transcriptions from them; and for text, we extract sentences from Wikipedia[5], WMT'14, and LibriSpeech LM data[6] (Panayotov et al., 2015).

4.2 System setups

To evaluate the generalization abilities of our methods across different languages, we conducted experiments on three S2ST directions, which are *German→English* (De–En), *French→English* (Fr–En), and *Spanish→English* (Es–En).

4.2.1 UASR

As indicated in Table 1, we used audio from CV4 and text data from Wikipedia to train our UASR models. More precisely, we use 100 hours of audio and about 1–3M sentences for each language. After training, we evaluate the results with the transcriptions from CV4. For the pre-trained W2V2 model, we directly use the cross-lingual version (XLSR; Conneau et al., 2020) without finetuning. XLSR had pre-trained in many different languages thus suiting our needs for training UASR in languages other than English.

During audio preprocessing, we apply 512 PCA dimension reduction, followed by mean-pooling based on k-means clustering ($K = 128$) and adjacent mean-pooling. For text preparation, we insert the <SIL> tokens at the rate of 0.25, except for French. We found that our French model converged better when <SIL> token insertion rate is 0.5 instead.

As for GAN training configuration, we chose the coefficients of the loss function according to the original paper as follows: the gradient penalty weight $\lambda = 1.5$ or 2.0, the smoothness penalty weight $\gamma = 0.5$, and the phoneme diversity loss weight $\eta = 4$. We trained 3 seeds for each configuration, conducting 6 models totally for each language.

For decoding, we apply all three strategies described in Section 3.1 for generating phoneme-level outputs. After that, we adopted self-training (ST) on HMM to further improve the performance and obtain word-level outputs.

4.2.2 UMT

For subword-level translation, we used the pre-trained English–German and English–French XLM released by Meta[7], and we collected 50M of monolingual Spanish, English texts from WMT'14 to train an English-Spanish XLM. We set all model to the same size (L = 6, H = 1024, A = 8, model size is about 798M)[8].

The pre-training process follows (Devlin et al., 2018) roughly, 15% of tokens are sampled from a sentence randomly, and 80% of the sampled tokens are replaced by [MASK], 10% are replaced by random tokens, and the remaining 10% are left unchanged.

For the back-translation, we extracted 1M to 3M sentences from Wikipedia[5] for each language, removed all punctuation marks, and normalized all characters to lowercase. The model is a seq2seq model, and both the encoder and decoder are initialized by the pre-trained XLM, the weight of auto-encoder loss linearly decreased from 1 to 0.1 in the first 10k steps, and linearly decreased from 0.1 to 0 in the following 20k steps.

[5] Extract the data using WikiExtractor (Attardi, 2015)

[6] Following Liu et al. (2022b), we exclude the transcriptions of LJspeech to form fully unpaired scenario

[7] https://github.com/facebookresearch/XLM

[8] L: numbers of transformer blocks. H: hidden size. A: numbers of attention heads.

Table 2: The results of our US2ST system on CVSS. Following Jia et al. (2022), the ST results are evaluated on CoVoST 2. C-ST stands for cascade ST; D-SS for direct S2ST.

Method	Type	ASR ↓			ST (X→En) ↑			S2ST (X→En) ↑		
		Fr	De	Es	Fr	De	Es	Fr	De	Es
SUPERVISED LEARNING										
(a) Wang et al. (2020b)	C-ST	18.3	21.4	16.0	27.6	21.0	27.4	-	-	-
(b) fairseq S2T (T-Sm) (Wang et al., 2020a)	D-ST	-	-	-	26.3	17.1	23.0	-	-	-
(c) fairseq S2T (Multi. T-Md)	D-ST	-	-	-	26.5	17.5	27.0	-	-	-
(d) Translatotron (Jia et al., 2019)	D-SS	-	-	-	-	-	-	15.5	6.9	14.1
(e) Translatotron 2 (Jia et al., 2021)	D-SS	-	-	-	-	-	-	28.3	19.7	23.5
(f) CVSS-C (ST→TTS) (Jia et al., 2022)	C-SS	-	-	-	31.9	23.9	33.9	31.2	23.9	33.3
(g) Our upper bound	C-SS	16.2	14.1	11.0	23.3	21.7	27.2	13.9	13.9	17.2
UNSUPERVISED LEARNING										
(h) Our cascade US2ST	C-SS	33.2	23.8	17.4	18.0	18.2	23.4	9.4	9.6	12.2

4.2.3 UTTS

To achieve UTTS, Ni et al. 2022 used Tactron2 for the acoustic modeling, while Liu et al. 2022b applied Transformer TTS for the acoustic modeling. And they both select HiFi-GAN as the vocoder. However, according to recent advances in TTS, Variational Inference with adversarial learning for end-to-end Text-to-Speech (VITS) has shown a significant performance gain over Tacotron2 and Transformer TTS in both subjective and objective evaluation(Kim et al., 2021; Hayashi et al., 2021). Therefore, in this paper, we employ VITS as our backbone TTS model.

For implementation, we utilize ESPnet-TTS to train the phonemicized text from UMT (Hayashi et al., 2020, 2021). The detailed configuration follow the LJSpeech recipe.[9]

4.3 Supervised cascade S2ST

We constructed our upper bound model by training a supervised cascade S2ST (ASR→MT→TTS) which shares similar model architecture with our US2ST.

For ASR, we finetune the whole XLSR instead of treating it as a feature extractor. We adopt letter-based training followed the whole configuration from fairseq[2] (Ott et al., 2019). The amount of audio data is exactly the same as those in UASR. Furthermore, we finetune the XLSR models individually for each language. MT is achieved by training the same seq2seq model with UMT, but the training data are the transcriptions of CVSS, and we randomly initialize the model, instead of using pre-trained XLM. The loss is the supervised MT loss, without adding the auto-encoder loss and back-translation loss. Finally, instead of training on pseudo labels from UASR, the supervised TTS model directly use the reference phonemes and their corresponding utterance as paired data.

By constructing cascade supervised S2ST, we can discuss the performance individually for each component.

4.4 Evaluation

The evaluation metric of ASR is word error rate (WER). We use the multi-bleu.perl script from Moses toolkit[10] to calculate the BLEU score of ST. For the final S2ST results, we use Whisper[11] (Radford et al., 2022), a supervised ASR model released by OpenAI, to transcribe the hypothesized audio, and calculate BLEU score.

4.5 Results

We show our overall results in Table 2, including the results of ASR, ST, and the final S2ST, indicating the performance in each stage of our cascade system. To compare our US2ST system with other supervised methods, we collect some of the results from the previous works on CoVoST 2 (*(a)–(c)*) and CVSS (*(d)–(f)*). Furthermore, to get a more complete comparison, both cascade and direct systems are included.

Next, we discuss the details of the methods in the table. In *(a)*, the results come from Wang et al. (2020b). Among all the experiments in their paper, we only report the results from the cascade ST constructed by mono-lingual ASR and bilingual MT. We consider the setup is the most suit-

[9] https://github.com/espnet/espnet/blob/master/egs2/ljspeech/tts1/conf/tuning/train_vits.yaml

[10] http://www2.statmt.org/moses/

[11] we use the English base model released by https://github.com/openai/whisper

able one against ours. According to the table, our US2ST performances in De–En and Es–En are just having small degradation from theirs (*(h)* vs *(a)*).

From *(b)* to *(c)*, we report the results of the direct ST systems from Wang et al. (2020a). They have developed a tool kit for ST and demonstrated it on CoVoST 2 with different model backbones. We compare our UST results (row *(h)*) with their Transformer-based models. Our results have not only outperformed their small model (*(b)*) in two translation directions but also shown comparable results with their larger multi-lingual model (row *(c)*).

From *(d)* to *(f)*, we present the S2ST results from the original paper of CVSS (Jia et al., 2022). As indicated in Table 2, our US2ST results might be comparable with the direct S2ST model, Translatotron (row *(f)*; Jia et al., 2019). However, their results came from literature, which might not be directly comparable due to using different ASR models for the S2ST evaluation. Finally, in row *(g)*, we show the results of our upper bound model, which is a supervised cascade S2ST system. Comparing with our upper bound method, our US2ST (row *(h)*) is just having a reasonable performance degradation across all the translation directions. In UST the results of US2ST are only about 4.2 BLEU score behind the upper bound in average. As for the US2ST, despite using the UTTS for systhesis, the performance gap just became slightly larger. In the three translation directions that we have implemented, all of them are having less than 5.0 BLEU score degradation against the supervised upper bound.

4.6 Analysis

In this section, we discuss and analyze more aspects and details of our cascade US2ST, showing how we choose between different sub-models and strategies.

Table 3: Stabilities of UASR across different languages.

Lang.	<SIL> ins. rate	Best PER (*Viterbi*)	%-converged (PER $< 50\%$)
De	0.25	25.3%	66%
Es	0.25	27.0%	50%
Fr	0.25	49.2%	<10%
	0.50	35.2%	17%

Table 4: Comparison of different decoding strategies and the improvement brought by HMM self-training. We use the same 4-gram LM (phoneme-level or word-level) across different methods.

Method	LM	PER(%)	WER(%)
(I) Without self-training			
Viterbi	✗	25.2	-
Kenlm	✓	29.5	39.5
WFST	✓	21.3	34.4
(II) With self-training			
Viterbi→ HMM	✓	15.2	25.3
WFST→ HMM	✓	**14.4**	**23.8**

Stabilities of UASR cross different languages First of all, we found that the stabilities of our UASR models vary between languages. The measurement of the stability is by calculating the percentage of the converged rate among the models under the same setting. We consider a UASR model is converged if its $PER < 50\%$. We summarize the discoveries in Table 3. According to our experiments, *German* and *Spanish* are easier to converge; while *French* usually can not converge well. However, we found that it might be better for *French* UASR models to converge if we change the <SIL> token insertion rate from 0.25 to 0.5.

Decoding and self-training in UASR The original outputs of wav2vec-U are in phoneme-level, which are incompatible with the UMT. However, with the integration with LM, we are available to obtain word-level output sequences. As shown in the part ***(I)*** of Table 4, we demonstrate that the two decoding methods, *Kenlm* and *WFST* can both generate word sequences by incorporating with phoneme-level or word-level LM. The second part ***(II)*** in the table illustrates the effectiveness of self-training on HMM. Among all the methods, we considered that the best strategy we found was by conducting self-training on HMM with the pseudo labels from *WFST* decoding. More surprisingly, even if the pseudo labels come from *Viterbi* decoding, using these labels on HMM can make huge improvements. After self-training, the performance gap between *Viterbi* and *WFST* decoding became relatively small. Note that for simplicity, we only show the results on the testing set of CV4-German; while the results on other languages also share similar trends.

Table 5: The UST and US2ST performance of using phoneme-level UMT.

Lang.	ref. PPL	UST PPL	US2ST BLEU
Fr	14.5	545.4	0.03
De	13.8	104.9	0.08
Es	13.4	14.7	0.10

Phoneme-level UMT Since the PER from UASR is lower than WER, besides the original setting, we also tried to perform our US2ST at phoneme level. In this setup, We followed the same training criterion of the subword-level UMT, but trained on a phonemicized sequence.

Direct calculating BLEU score on the phoneme sequence generated by UST is unmeaningful. The number of phonemes is greatly less than that of words, so randomly generating some phoneme sequences can still get a reasonable score. As a result, we turn to test the naturalness of generated phonemes, we trained a phoneme-level 4-gram language model, and calculate the perplexity of UST results.

The results are shown in Table 5. The models of French and German are unable to generate a natural English phoneme sequence; thus the US2ST also failed. Although the Spanish model can generate natural English phoneme sequences, the performance of US2ST is still terrible, which implies the model did not retain the original meaning of the input.

The phoneme-level UMT might be too hard: word boundary information is missing in the phoneme sequence, so the model should find the word boundary between phonemes, and tries to translate the words to English. Such difficulty might prevent the phoneme-level UMT from working well.

Table 6: Analysis of our UTTS models. We use WER as the evaluation metric.

Input phn.	TTS	UTTS train on diff. PER	
	sup.	$per = 22\%$	$per = 14\%$
(I) Using phonemicized or UASR-generated phn.			
Phonemicized	23.5%	37.7%	31.5%
UASR-generated	-	40.4%	28.9%
(II) Using phonemicized UST results			
Fr→En	46.5%	61.9%	54.2%
De→En	38.6%	56.0%	47.0%
Es→En	43.3%	59.2%	50.2%

Analysis on UTTS First of all, to get more insights about the impact of using different level of PER for training UTTS, we train two UTTS models with different PER; furthermore, a supervised TTS are trained to form the upper bound, as shown in Table 6. Initially, to validate our TTS models, we just feed the phonemicized sequences from reference sentences into the models and calculate WER after transcribing the generated waveform. However, since all of the UTTS models have never seen the real, or the phonemicized sequences before, we are wondering if the UTTS models can perform better when taking sequences from their corresponding UASR as input. Thus, in the section *(I)* in Table 6, we aim to analyze the issue by taking either phonemicized or generated sequences as the inputs for UTTS. Interestingly, we found that in some cases the UTTS models actually performed better with their corresponding UASR-generated sequences.

Secondly, in section *(II)* of the table, we measure the performance degradation by showing the detailed WER of our UTTS models taking as input from UST. Although the WER of UTTS models are higher than the supervised upper bound, the performance degradation on supervised TTS is also severe, indicating the impact of mismatch between the input data for the TTS models.

5 Conclusions

In this paper, we proposed a cascade US2ST (unsupervised speech-to-speech translation) system, the training process does not rely on any labeled data, and the performance can be comparable with or even outperform supervised works in some cases. In addition, we analyzed how languages and decoding strategies influence the performance of UASR and UTTS.

While there is still a big gap between our work and the SOTA supervised S2ST system. Inspired by recent success in the self-supervised speech pre-trained model, our future work includes translating between discrete tokens generated by the self-supervised pre-trained model in an unsupervised scenario (Tjandra et al., 2019), and we expect that leveraging representations from self-supervised model to build US2ST system can mitigate the gap.

References

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*.

Giusepppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Alexei Baevski, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. 2021. Unsupervised speech recognition. *Advances in Neural Information Processing Systems*, 34:27826–27839.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460.

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*.

Mathieu Bernard and Hadrien Titeux. 2021. Phonemizer: Text to phones transcription for multiple languages in python. *Journal of Open Source Software*, 6(68):3958.

Hailong Cao, Tiejun Zhao, Shu Zhang, and Yao Meng. 2016. A distribution-based model to learn bilingual word embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1818–1827.

Kuan-Yu Chen, Che-Ping Tsai, Da-Rong Liu, Hung-Yi Lee, and Lin-shan Lee. 2019. Completely unsupervised speech recognition by a generative adversarial network harmonized with iteratively refined hidden markov models. *arXiv preprint arXiv:1904.04100*.

Yu-An Chung, Yuxuan Wang, Wei-Ning Hsu, Yu Zhang, and RJ Skerry-Ryan. 2019a. Semi-supervised training for improving data efficiency in end-to-end speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6940–6944. IEEE.

Yu-An Chung, Wei-Hung Weng, Schrasing Tong, and James Glass. 2018. Unsupervised cross-modal alignment of speech and text embedding spaces. *Advances in neural information processing systems*, 31.

Yu-An Chung, Wei-Hung Weng, Schrasing Tong, and James Glass. 2019b. Towards unsupervised speech-to-text translation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7170–7174. IEEE.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2020. Unsupervised cross-lingual representation learning for speech recognition. *arXiv preprint arXiv:2006.13979*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mattia Antonino Di Gangi, Matteo Negri, and Marco Turchi. 2019. Adapting transformer to end-to-end spoken language translation. In *INTERSPEECH*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, Tomoki Toda, Kazuya Takeda, Yu Zhang, and Xu Tan. 2020. Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 7654–7658. IEEE.

Tomoki Hayashi, Ryuichi Yamamoto, Takenori Yoshimura, Peter Wu, Jiatong Shi, Takaaki Saeki, Yooncheol Ju, Yusuke Yasuda, Shinnosuke Takamichi, and Shinji Watanabe. 2021. Espnet2-tts: Extending the edge of tts research. *arXiv preprint arXiv:2110.07840*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Hirofumi Inaguma, Kevin Duh, Tatsuya Kawahara, and Shinji Watanabe. 2019. Multilingual end-to-end speech translation. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 570–577. IEEE.

Keith Ito and Linda Johnson. 2017. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/.

Ye Jia, Michelle Tadmor Ramanovich, Tal Remez, and Roi Pomerantz. 2021. Translatotron 2: Robust direct speech-to-speech translation. *arXiv preprint arXiv:2107.08661*.

Ye Jia, Michelle Tadmor Ramanovich, Quan Wang, and Heiga Zen. 2022. Cvss corpus and massively multilingual speech-to-speech translation. *arXiv preprint arXiv:2201.03713*.

Ye Jia, Ron J Weiss, Fadi Biadsy, Wolfgang Macherey, Melvin Johnson, Zhifeng Chen, and Yonghui Wu. 2019. Direct speech-to-speech translation with a sequence-to-sequence model. *arXiv preprint arXiv:1904.06037*.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. *arXiv preprint arXiv:1804.07755*.

Alon Lavie, Alex Waibel, Lori Levin, Michael Finke, Donna Gates, Marsal Gavalda, Torsten Zeppenfeld, and Puming Zhan. 1997. Janus-iii: Speech-to-speech translation in multiple languages. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 99–102. IEEE.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, Juan Pino, et al. 2021. Direct speech-to-speech translation with discrete units. *arXiv preprint arXiv:2107.05604*.

Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2020. Multilingual speech translation with efficient finetuning of pretrained models. *arXiv preprint arXiv:2010.12829*.

Jiachen Lian, Chunlei Zhang, Gopala Krishna Anumanchipalli, and Dong Yu. 2022. Utts: Unsupervised tts with conditional disentangled sequential variational auto-encoder. *arXiv preprint arXiv:2206.02512*.

Guan-Ting Lin, Chan-Jan Hsu, Da-Rong Liu, Hung-Yi Lee, and Yu Tsao. 2022. Analyzing the robustness of unsupervised speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8202–8206. IEEE.

Alexander H Liu, Wei-Ning Hsu, Michael Auli, and Alexei Baevski. 2022a. Towards end-to-end unsupervised speech recognition. *arXiv preprint arXiv:2204.02492*.

Alexander H Liu, Cheng-I Jeff Lai, Wei-Ning Hsu, Michael Auli, Alexei Baevskiv, and James Glass. 2022b. Simple and effective unsupervised speech synthesis. *arXiv preprint arXiv:2204.02524*.

Da-Rong Liu, Kuan-Yu Chen, Hung-yi Lee, and Lin-shan Lee. 2018. Completely unsupervised phoneme recognition by adversarially learning mapping relationships from audio embeddings. *arXiv preprint arXiv:1804.00316*.

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. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Mehryar Mohri, Fernando Pereira, and Michael Riley. 2002. Weighted finite-state transducers in speech recognition. *Computer Speech & Language*, 16(1):69–88.

Satoshi Nakamura, Konstantin Markov, Hiromi Nakaiwa, Gen-ichiro Kikui, Hisashi Kawai, Takatoshi Jitsuhiro, J-S Zhang, Hirofumi Yamamoto, Eiichiro Sumita, and Seiichi Yamamoto. 2006. The atr multilingual speech-to-speech translation system. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):365–376.

Junrui Ni, Liming Wang, Heting Gao, Kaizhi Qian, Yang Zhang, Shiyu Chang, and Mark Hasegawa-Johnson. 2022. Unsupervised text-to-speech synthesis by unsupervised automatic speech recognition. *arXiv preprint arXiv:2203.15796*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev 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. IEEE.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. 2021. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921. IEEE.

Lawrence Rabiner and Biinghwang Juang. 1986. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. Technical report, Technical report, OpenAI, 2022. URL https://cdn. openai. com/papers/whisper. pdf.

Yi Ren, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2019. Almost unsupervised text to speech and automatic speech recognition. In *International Conference on Machine Learning*, pages 5410–5419. PMLR.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015a. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015b. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Zheng-Hua Tan, Najim Dehak, et al. 2020. rvad: An unsupervised segment-based robust voice activity detection method. *Computer speech & language*, 59:1–21.

Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura. 2019. Speech-to-speech translation between untranscribed unknown languages. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 593–600. IEEE.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103.

Wolfgang Wahlster. 2000. Verbmobil: Foundations of speech-to-speech translation. In *Artificial Intelligence*.

Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Dmytro Okhonko, and Juan Pino. 2020a. fairseq s2t: Fast speech-to-text modeling with fairseq. *arXiv preprint arXiv:2010.05171*.

Changhan Wang, Anne Wu, and Juan Pino. 2020b. Covost 2 and massively multilingual speech-to-text translation. *arXiv preprint arXiv:2007.10310*.

Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. *arXiv preprint arXiv:1703.08581*.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.

Chen Zhang, Xu Tan, Yi Ren, Tao Qin, Kejun Zhang, and Tie-Yan Liu. 2021. Uwspeech: Speech to speech translation for unwritten languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14319–14327.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970.