Leveraging Unit Language Guidance to Advance Speech Modeling in Textless Speech-to-Speech Translation

Yuhao Zhang[1,2], Xiangnan Ma[1], Kaiqi Kou[1], Peizhuo Liu[1], Weiqiao Shan[1],
Benyou Wang[2], Tong Xiao[1,4*], Yuxin Huang[3], Zhengtao Yu[3] and Jingbo Zhu[1,4]
[1] Northeastern University, Shenyang, China
[2] The Chinese University of Hong Kong, Shenzhen, China
[3] Kunming University of Science and Technology, Kunming, China
[4]NiuTrans Research, Shenyang, China
yoohao.zhang@gmail.com, {xiaotong, zhujingbo}@mail.neu.edu.cn

Abstract

The success of building textless speech-to-speech translation (S2ST) models has attracted much attention. However, S2ST still faces two main challenges: 1) extracting linguistic features for various speech signals, called cross-modal (CM), and 2) learning alignment of difference languages in long sequences, called cross-lingual (CL). We propose the unit language to overcome the two modeling challenges. The unit language can be considered a text-like representation format, constructed using n-gram language modeling. We implement multi-task learning to utilize the unit language in guiding the speech modeling process. Our initial results reveal a conflict when applying source and target unit languages simultaneously. We propose task prompt modeling to mitigate this conflict. We conduct experiments on four languages of the Voxpupil dataset. Our method demonstrates significant improvements over a strong baseline and achieves performance comparable to models trained with text.

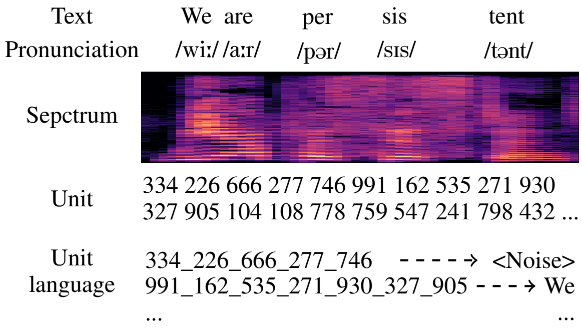


Figure 1: The proposed unit language and three other text types represent the transcription. The unit language is created using unsupervised language modeling.

1 Introduction

The Speech-to-Speech Translation (S2ST) task aims to generate target speech according to the source speech, which can significantly improve communication efficiency between speakers of different languages. Conventional methods apply the cascade method that uses automatic speech recognition (ASR), machine translation (MT), and text-to-speech (TTS) models (Vidal, 1997; Casacuberta et al., 2004; Aguero et al., 2006). This strategy always suffers from error propagation and high latency, thus researchers turn to investigate direct S2ST (Jia et al., 2019; Lee et al., 2022a; Inaguma et al., 2023). The direct S2ST models the source speech and generates the unit or spectrum of the target speech within one model. Inspired by the direct S2ST paradigm, Lee et al. (2022b) proposed *textless* S2ST, namely achieving speech-to-speech transformation with the constraint of unit or VQ-VAE token. This method eliminates the need for any labeled text and is very valuable for languages without text-writing systems.

However, there are still two challenging problems for textless S2ST (Lee et al., 2022a; Jia et al., 2021): 1) how to learn the acoustic and linguistic features from the varying and continuous audio signal without transcription (calling it cross-modal modeling, CM), and 2) how to achieve alignment between languages along the long sequence without translated text pairs (calling it cross-lingual modeling, CL). For the former problem, previous works propose the quantization method to improve the discrete unit or token, aiming to be more suitable for cross-modal and cross-lingual modeling (Zhang et al., 2020; Lee et al., 2022a; Li et al., 2023). In the latter case, some masked language models or denoising auto-encoder methods were used to improve cross-lingual modeling (Popuri et al., 2022; Chen et al., 2023). Though these methods show improvement in textless S2ST, the CL and CM challenges have not been adequately addressed due to the lack of guidance from text.

We design a format called the unit language, serving as a transcription of speech to enhance CM

[*] Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2025, pages 1448–1460
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

and CL modeling. The unit language consists of unit words which are merged from several contiguous speech units. Each unit word is generated based on n-gram language modeling. This strategy searches for the maximum probability of unit words for every unit sequence and does not rely on any labeled data. This novel representation can serve as an alternative to text during the modeling process, addressing the challenges posed by textless training. As shown in Figure 1, the unit language is implicitly aligned with the real text. We further implement multi-task learning based on the source and target unit language to guide CM and CL modeling respectively.

We conducted experiments on the Voxpupil dataset (Wang et al., 2021). Our method achieved an average improvement of 1.2 BLEU over the strong baseline. Furthermore, it demonstrates performance comparable to the S2ST model trained with text. This shows that the unit language can serve as an effective alternative to text in the speech modeling process. Our further analysis reveals that CM and CL processing based on unit language have distinct impacts, but both are essential for S2ST. CM processing filters noise in speech, while CL processing helps capture semantic information for translation. However, the negative impact of CL on the effectiveness of CM prevents the simultaneous application of both methods from achieving consistent improvements. To address this issue, we proposed task prompt modeling to mitigate the conflict. Final results demonstrate that our model achieves new state-of-the-art performance on the textless S2ST task using the Voxpupil dataset[1].

2 Method

The philosophy of this paper is to utilize a text-like format, namely the unit language, to enhance speech modeling. To use the unit language, we introduce two additional decoders and employ multi-task learning to guide the modeling process. We propose task prompt modeling as a mitigation strategy to address task conflicts in multi-task learning.

2.1 Unit Language Construction

We propose a method to construct the unit language based on language modeling processing. Formally, given a unit sequence $\{u_1, u_2, ..., u_n\}$, our goal is to convert it into a word sequence $\{w_1, w_2, ..., w_m\}$, where each w consists of at most K continuous units and K is a hyper-parameter. Considering an arbitrary sub-sequence $u_{[1:i]} = \{u_1, u_2, ..., u_i\}$, we can generate a corresponding sequence $w_{[1:j]} = \{w_1, w_2, ..., w_j\}$. Considering the original unit can be various, we apply the **norm unit** (Lee et al., 2022b), which is produced by a model trained with data from a specific speaker. The norm unit is less noisy and easier to learn. According to the process of language modeling, the probability of $w_{[1:j]}$ can be calculated by:

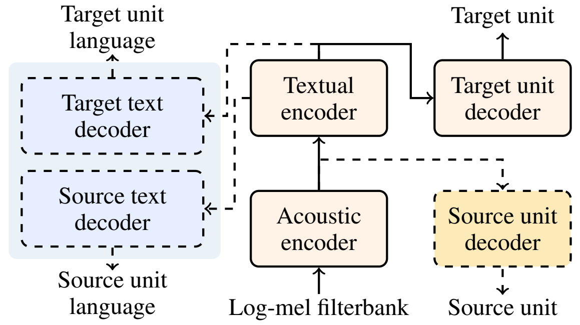


Figure 2: Architecture of textless S2ST. The dashed line part will be removed during inference. The filterbank features serves as the only input to the network. Blue modules are used to guide the modeling process.

$$P(w_{[1:j]}) = P(w_1)P(w_2|w_1)...P(w_j|w_1...w_{j-1}). \quad (1)$$

When the $P(w_{[1:j]})$ reaches its maximum, which is the maximum likelihood, we can consider $w_{[1:j]}$ as the optimal unit language sequence to represent $u_{[1:i]}$. We use $\pi(u_{[1:i]})$ to denote the optimal conversion path as follows:

$$\pi(u_{[1:i]}) = w^*_{[1:j]} = \arg\max_{w_{[1:j]}}(P(w_{[1:j]})). \quad (2)$$

1-gram Our final goal is to find π for any unit sequence. To make the deduction process clear, we first consider the circumstance of conditional independence, namely 1-gram. Then we can get:

$$\begin{aligned} P(w_{[1:j]}) &\approx P(w_1)P(w_2)...P(w_j) \\ &= \exp(\sum_t^j \log P(w_t)). \end{aligned} \quad (3)$$

Considering that w_j consists of a maximum of K continuous units, we can derive the following re-

[1] Code: https://github.com/xiaozhang521/Unit_Language

cursion formula for w_j and $w_{[1:j-1]}$:

$$\max(\sum_{t}^{j} \log P(w_t)) =$$
$$\max \begin{Bmatrix} \log P(\pi(u_{[1:i-1]})) + \log P(u_i), \\ \log P(\pi(u_{[1:i-2]})) + \log P(u_{[i-1,i]}), \\ ... \\ \log P(\pi(u_{[1:i-K]})) + \log P(u_{[i-K+1,i]}) \end{Bmatrix}. \tag{4}$$

For any $k \leq K$, we can use the frequency from the unit corpus to estimate $P(u_{[i-k+1:i]})$. Then, we get k_i^* as the maximum element:

$$k_i^* = \arg\max_k(\log P(\underbrace{\pi(u_{[1:i-k]})}_{w^*_{[1:j-1]}}) + \log P(\underbrace{u_{[i-k+1,i]}}_{w_j})). \tag{5}$$

For position i, we can determine w_j using k_i^* units, namely $w_j = u_{[i-k_i^*+1,i]}$. Then w_{j-1} can be determined by $k^*_{i-k_i^*}$. After applying dynamic programming, the entire sequence $w^*_{[1:m]}$ can be obtained for any unit sequence $u_{[1:n]}$, and we consider the generated sequence $w^*_{[1:m]}$ as the unit language.

2-gram For most languages, considering in-context information is necessary, and we can easily adapt our method to n-gram modeling. Considering the computational cost (refer to Appendix), we primarily use 2-gram modeling in this paper. Thus, equation (2) can be updated as follows:

$$\begin{aligned} \pi(u_{[1:i]}) &= \arg\max_{w_{[1:j]}}(P(w_{[1:j]})) \\ &= \arg\max_{w_{[1:j]}}(\sum_{t>1}^{j} \log P(w_t|w_{t-1}) + \log P(w_1)) \end{aligned} \tag{6}$$

Similarly, we can update the formula (4) as follows:

$$\max(\sum_{t>1}^{j} \log P(w_t|w_{t-1})) =$$
$$\max \begin{Bmatrix} \log P(\pi(u_{[1:i-1]})) \\ + \log P(u_i|u_{[i-k^*_{i-1}:i-1]}), \\ \log P(\pi(u_{[1:i-2]})) \\ + \log P(u_{[i-1:i]}|u_{[i-k^*_{i-2}-1:i-2]}), \\ ... \\ \log P(\pi(u_{[1:i-K]})) \\ + \log P(u_{[i-K+1:i]}|u_{[i-k^*_{i-K}-K+1:i-K]}) \end{Bmatrix} \tag{7}$$

Loss	Output	Training modules
$\mathcal{L}_{\text{SU}}$	Source unit	{A-Enc, SU-Dec}
$\mathcal{L}_{\text{TU}}$	Target unit	{A-Enc, T-Enc, TU-Dec}
$\mathcal{L}_{\text{CM}}$	Source unit language	{A-Enc, T-Enc*, S-Dec}
$\mathcal{L}_{\text{CL}}$	Target unit language	{A-Enc, T-Enc, T-Dec}
$\mathcal{L}_{\text{CM}'}$	Source text	{A-Enc, T-Enc*, S-Dec}
$\mathcal{L}_{\text{CL}'}$	Target text	{A-Enc, T-Enc, T-Dec}

Table 1: Overview of all the tasks and modules. The input is filterbank for all the tasks. * denotes that the loss updates parts of parameters in the module.

For any $k \leq K$ and $l = k^*_{i-k} - k$, we still use the frequency to estimate the conditional probability as follows:

$$P(u_{[i-k+1:i]}|u_{[i-l+1:i-k]}) = \frac{P(u_{[i-l+1:i]})}{P(u_{[i-l+1:i-k]})}. \tag{8}$$

2.2 Architecture

We improve the architecture of advanced textless S2ST model (Lee et al., 2022b) as shown in Figure 2. It converts source audio a_s and target audio a_t to discrete units u_s and u_t, respectively. The source units u_s and target units u_t are produced by the pre-trained multilingual Hubert model (Hsu et al., 2021), and u_s and u_t serve as pseudo source and target text. We apply the norm unit (Lee et al., 2022b) as the training target to achieve strong baselines.

To clarify the architecture, we define the baseline as consisting of four parts: acoustic encoder (A-Enc), textual encoder (T-Enc), source unit decoder (SU-Dec), and target unit decoder (TU-Dec). We add two additional decoders to process source and target unit languages. Since both decoders generate the text-like unit language, we call them the source text decoder (S-Dec) and target text decoder (T-Dec), respectively. Previous analysis shows that the representation is converted from speech to text in the textual encoder (Zhang et al., 2023). We choose the intermediate state of T-Enc as the input for S-Dec and T-Dec.

At the inference stage, the audio features are fed into the A-Enc. After processing by A-Enc and T-Enc, the TU-Dec directly generates target units, which are subsequently passed to a vocoder to synthesize the target speech.

2.3 Multi-task Learning

Training the textless S2ST model is challenging, and mainstream methods rely on multi-task learning. Specifically, the input is the filterbank feature

Models	Es-En	Fr-En	En-Es	En-Fr	Avg.
Seamless (Barrault et al., 2023)	34.7	35.9	29.2	25.0	31.2
Textless S2UT (Lee et al., 2022b)	18.9	19.9	22.7	18.7	20.1
S2UT* (Lee et al., 2022b)	19.4	19.7	21.8	18.9	20.0
w/ Recognized text					
Baseline (+ $\mathcal{L}_{\text{TU}}$&$\mathcal{L}_{\text{SU}}$)	19.1	20.3	23.0	18.8	20.3
+ $\mathcal{L}_{\text{CM}'}$	19.7 (+0.6)	21.0 (+0.7)	23.9 (+0.9)	**20.8** (+2.0)	21.4 (+1.1)
+ $\mathcal{L}_{\text{CL}'}$	**20.3** (+1.2)	**21.2** (+0.9)	23.9 (+0.9)	20.6 (+1.8)	**21.5** (+1.2)
+ $\mathcal{L}_{\text{CM}'}$&$\mathcal{L}_{\text{CL}'}$	19.8 (+0.7)	20.8 (+0.5)	**24.0** (+1.0)	**20.8** (+2.0)	21.4 (+1.1)
w/ Unit language					
Baseline (+ $\mathcal{L}_{\text{TU}}$&$\mathcal{L}_{\text{SU}}$)	19.1	20.3	23.0	18.8	20.3
+ $\mathcal{L}_{\text{CM}}$	19.2 (+0.1)	20.7 (+0.4)	23.5 (+0.5)	19.5 (+0.7)	20.7 (+0.4)
+ $\mathcal{L}_{\text{CL}}$	19.4 (+0.3)	21.0 (+0.7)	23.9 (+0.9)	19.9 (+1.1)	21.1 (+0.8)
+ $\mathcal{L}_{\text{CM}}$&$\mathcal{L}_{\text{CL}}$	19.7 (+0.6)	21.0 (+0.7)	23.8 (+0.8)	20.4 (+1.6)	21.2 (+0.9)
+Task prompt	**19.9** (+0.8)	**21.1** (+0.8)	**24.4** (+1.4)	**20.6** (+1.8)	**21.5** (+1.2)

Table 2: Performance on different datasets. Our baseline is a reproduction (Lee et al., 2022b). * denotes that the model does not use the norm unit.

of a_s, and the output of the acoustic encoder is used to predict the source unit u_s as the training loss:

$$\mathcal{L}_{\text{SU}} = -\log P(u_s|a_s, \theta_{\text{A-Enc}}, \theta_{\text{SU-Dec}}). \quad (9)$$

The output of the whole encoder (A-Enc and T-Enc) is used to predict the target unit by TU-Dec. The cross-lingual loss is denoted as:

$$\mathcal{L}_{\text{TU}} = -\log P(u_t|a_s, \theta_{\text{A-Enc}}, \theta_{\text{T-Enc}}, \theta_{\text{TU-Dec}}). \quad (10)$$

To improve the effect of CM, we use the unit language as the transcription of each source audio, denoted $\widetilde{u}_s$. The S-Dec processes the output of the r-th layer in T-Enc, where r is a hyperparameter. The auxiliary CM loss can be denoted as follows.

$$\mathcal{L}_{\text{CM}} = -\log P(\widetilde{u}_s|a_s, \theta_{\text{A-Enc}}, \theta^{r}_{\text{T-Enc}}, \theta_{\text{S-Dec}}) \quad (11)$$

where $\theta^{r}_{\text{T-Enc}}$ indicates that the parameters before the r-th layer in T-Enc are used. Similarly, the target unit language $\widetilde{u}_t$ can be generated according to u_t. We view $\widetilde{u}_t$ as translation text and implement the T-Dec to compute the auxiliary CL loss:

$$\mathcal{L}_{\text{CL}} = -\log P(\widetilde{u}_t|a_s, \theta_{\text{A-Enc}}, \theta_{\text{T-Enc}}, \theta_{\text{T-Dec}}). \quad (12)$$

We combine all the training losses to form the final training objective:

$$\mathcal{L} = \mathcal{L}_{\text{TU}} + \alpha\mathcal{L}_{\text{SU}} + \beta\mathcal{L}_{\text{CM}} + \gamma\mathcal{L}_{\text{CL}} \quad (13)$$

where α is fixed at 8. β and γ are set to 8 if the corresponding task is activated. We also test the multi-task learning approach by applying text as the auxiliary data. The losses $\mathcal{L}_{\text{CM}'}$ and $\mathcal{L}_{\text{CL}'}$ represent the scenarios where we replace the unit language with the source and target text, respectively. Table 1 provides brief information about terms of multi-task learning.

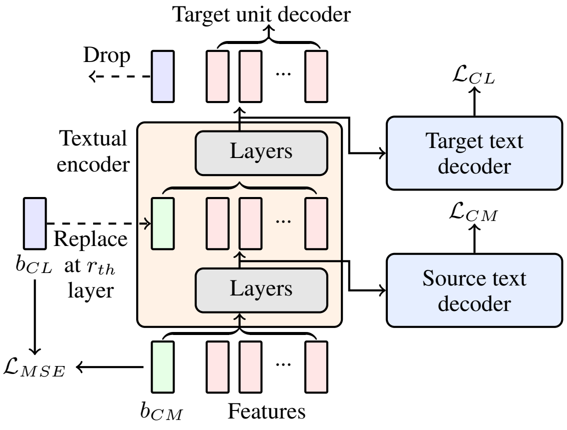


Figure 3: The guiding process of the task prompt.

2.4 Task Prompt Modeling

In our subsequent analysis, we observe a conflict between $\mathcal{L}_{\text{CM}}$ and $\mathcal{L}_{\text{CL}}$. To address this issue, we introduce task prompts to improve multi-task learning when applying the two tasks simultaneously. Specifically, we employ two learnable weights as task prompts, namely $b_{\text{CM}} \in \mathbb{R}^{1\times h}$ and $b_{\text{CL}} \in \mathbb{R}^{1\times h}$, where h is the hidden size.

We use b_{CM} and b_{CL} as inductive biases for the CM and CL tasks respectively. The overall processing is shown in Figure 3. Before the features are fed into the T-Enc, the prompt b_{CM} is concatenated with the features at the first position. The interme-

we are human beings.
704_334 365_548_985 99_991 535_271_930 (we) 327_905_579 (are) 933_901 427_258_436 139_340_748 872_336_877 (human) 488_620_915 143_290_978 485_113 (be) 398_212_455 545_711_510 (ings) 337_243 59
we are relatives.
681_63 991_162 535_271_930 (we) 327_905_579 (are) 969_156_824 384_879 259_317_453 275_830_471 737_53_885 545_85_510 (relatives) 297_206_265

Table 3: Comparison between unit language and text. The gray unit language denotes the noise token.

diate features of the r-th layer are used to compute $\mathcal{L}_{\text{CM}}$. Then the task prompt $\mathcal{L}_{\text{CM}}$ is replaced by b_{CL} after the r-th layer to adapt the CL modeling. Furthermore, we add an extra loss, calculated by the mean square error between b_{CM} and b_{CL} with a negative weight of -3.0. This enhances the diversity of the prompts and leads to improved performance.

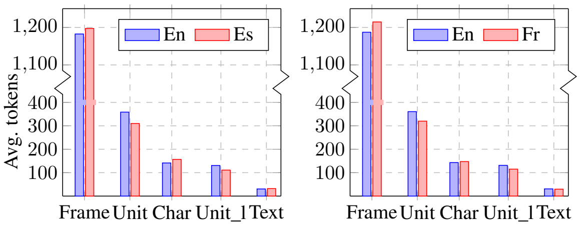
3 Experiments

3.1 Data and Model Settings

We conducted experiments on the VoxPopuli speech-to-speech dataset (Wang et al., 2021). The source and target units are converted by mHubert (Lee et al., 2022b) with 1000 units. The input speech features are 80-dimensional filter banks. The normalized unit is generated by the speaker normalizer (Lee et al., 2022b). We set K to 3 to search for the unit language for all translation pairs. We use SentencePiece (Kudo and Richardson, 2018) to control the vocabulary size to 10k. For S2ST training with text, we utilize pre-trained ASR models[2] and split the words to the character level as the training target.

We use the Transformer (Vaswani et al., 2017) as the backbone network with a 12-layer encoder and a 6-layer decoder. The hidden size is 512. The unit representation is the output of the 6th layer. The source and target text decoders are set to 2 layers. The parameter r is set to 2, meaning the source decoders use the output of the 2nd T-Enc layer. The target decoders are implemented after the T-Enc, which is the 12th layer of the whole encoder. The vocoder we used is the unit-based HiFi-GAN (Kong et al., 2020; Polyak et al., 2021).

During the training stage, we adopt a different strategy, utilizing a larger learning rate (0.001) and a bigger batch size (40,000 tokens). This approach enables our models to converge faster and achieve better performance. All experiments are conducted on 8 RTX 3090 GPUs. We use the best checkpoint for evaluation. Pre-trained ASR models[1] are used to recognize speech generated by the S2ST model. To normalize references, we remove punctuation, convert numbers to spoken forms, and lowercase text, following the work of Lee et al. (2022b). We report ASR SacreBLEU (Post, 2018). More training and data details can be found in the Appendix.



(a) En-Es

(b) En-Fr

Figure 4: Average lengths of different types of tokens on En-Es and En-Fr training datasets. "Unit_l" denotes the unit language. Note that the units used here have had continuous repetitions removed.

3.2 Results

We compare the advanced textless S2ST with the text-based model (Barrault et al., 2023) in Table 2. This comparison shows that the textless model has significant improvement potential. After applying unit language as the auxiliary training data, results in Table 2 show that both $\mathcal{L}_{\text{CL}}$ and $\mathcal{L}_{\text{CM}}$ improve the performance of the textless model. The two losses achieve average improvements of 0.4 to 0.8 BLEU based on strong baselines. All the translation pairs show consistent improvement. We find that the improvement gained from CL training is greater than that from CM training, which indicates that the textless S2ST requires crucial enhancement in cross-lingual learning.

We further compare our method with a text-based method. The results also prove that both source and target texts have much potential to improve the S2UT. We find that our method shows al-

[2] En: https://huggingface.co/facebook/wav2vec2-large-960h-lv60-self, Es: https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-spanish, Fr: https://huggingface.co/jonatasgrosman/wav2vec2-large-fr-voxpopuli-french

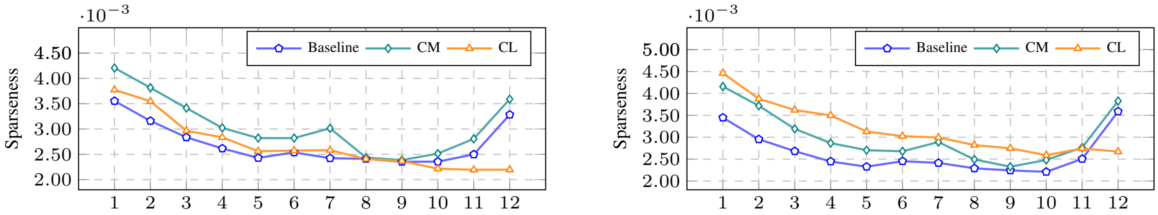


Figure 5: The influences of CM and CL on Fr-En (left) and En-Fr (right) tasks.

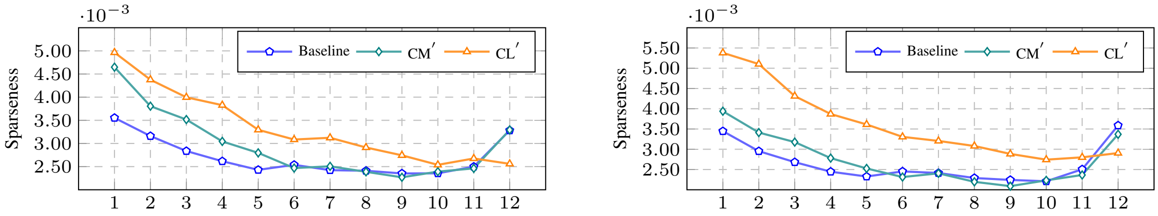


Figure 6: The influences of CM$'$ and CL$'$ on Fr-En (left) and En-Fr (right) tasks.

most comparable performance with the text-based method in cross-lingual learning. This demonstrates that our unit language, mined through language modeling, can effectively function like text.

When we apply the $\mathcal{L}_{\mathrm{CL}'}$ and $\mathcal{L}_{\mathrm{CM}'}$ methods simultaneously, the performance does not further increase compared with either method alone, and even degradation occurs. The Fr-En and En-Es pairs trained with the unit language confirm this phenomenon. This indicates that the effects of the two methods may not be the same and that conflicts have occurred during the modeling process. We aim to isolate the effects of $\mathcal{L}_{\mathrm{CL}}$ and $\mathcal{L}_{\mathrm{CM}}$ by adding task prompts method. Results in Table 2 demonstrate that this approach harmonizes CM and CL methods, achieving much advanced performance in all textless S2ST directions. Furthermore, our unit language can help the model achieve performance comparable to true text, which could significantly aid languages without sufficient labeled text for modeling speech.

4 Analysis

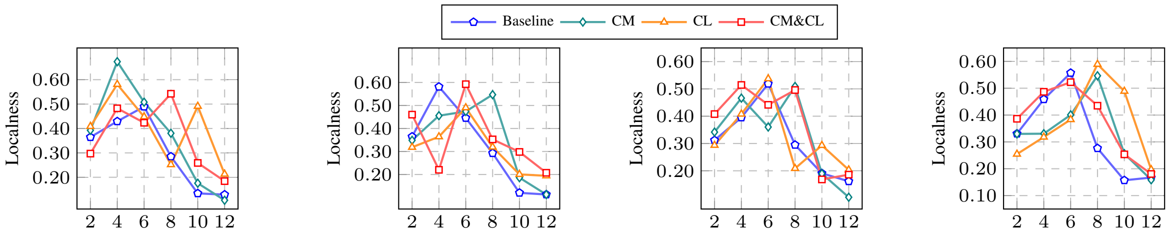
4.1 Comparison of Unit Language and Text

Since the unit sequence is rather long, we choose some short speech samples for a case study to compare with text. As shown in Table 3, the unit language is basically merged according to pronunciation. Thus, the size of the unit language representing a word depends on the syllables of the word. Furthermore, even across different sentences, the unit language can successfully represent the same word. Concise sequences benefit the extraction of important information (Chan et al., 2015), thus the unit language could address the CL and CM modeling challenges in textless S2ST.

We present the average length of different types of training data in Figure 4. It is very difficult to learn cross-lingual alignment based on thousands of various frames if there is no intermediate guidance. Units could help with modeling speech and they are several times shorter than frames (Lee et al., 2022b). But units still have an obvious inconsistency in length compared with characters or text as shown in Figure 4. Previous work suggests that length reduction benefits cross-lingual modeling in speech translation (Zhang et al., 2023). After applying language modeling, we find that the length of the unit language is between that of characters and text, making it suitable for learning cross-lingual alignment. Furthermore, we found that unit language is stable across different languages and consistently shows significant compression of speech sequences. This demonstrates that unit language has great potential for speech-related tasks, which could boost CM and CL learning.

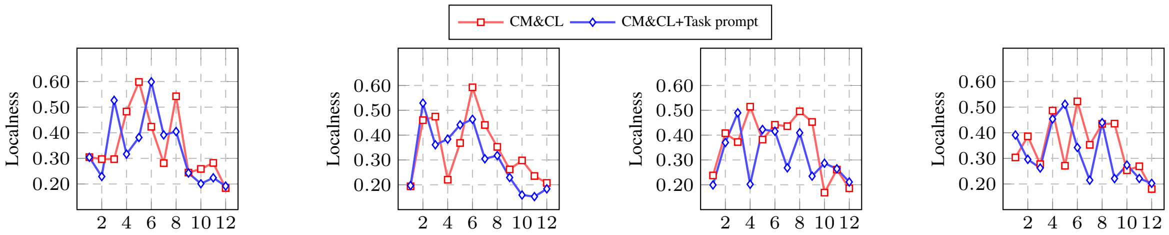
4.2 How CM and CL Work?

We have observed that both $\mathcal{L}_{\mathrm{CM}}$ and $\mathcal{L}_{\mathrm{CL}}$ contribute to textless S2ST, then we reveal how these two losses affect the model. We sample 200 audios from the training dataset to compute the *Sparseness* metric and analyze the effect on S2ST models. Sparseness is calculated by determining the proportion of values with absolute values less than 1e-3 in the representations. The representations are extracted from the normalized output of each encoder layer. This metric mainly measures the number



(a) Es-En (b) Fr-En (c) En-Es (d) En-Fr

Figure 7: Localness of attention weight on different tasks.



(a) Es-En (b) Fr-En (c) En-Es (d) En-Fr

Figure 8: Localness of attention weight with/without task prompt on different language tasks.

of non-activated nodes in the representation, with higher sparseness indicating more unnecessary information. To simplify the expression, we use CM and CL to denote $\mathcal{L}_{\text{CM}}$ and $\mathcal{L}_{\text{CL}}$ separately in the later analysis section. Similarly, $\text{CM}^{'}$ and $\text{CL}^{'}$ denote models trained based on the recognized text.

Figure 5 illustrates the Sparseness of different layers resulting from different methods. We observe distinct trends for CL and CM. Compared with middle layers, Baseline and CM exhibit higher Sparseness in earlier and top layers. This shows more nodes are not needed for the bottom encoder and target decoder, suggesting its effectiveness in noise filtering but limited cross-lingual understanding. The difference is the slight fluctuation in the 7-th layer for CM. After applying CL, the representation learned other information which differs from the source unit. Conversely, CL does not show an increase in Sparseness in top layers, indicating its better cross-lingual capability.

We test the model Sparseness after applying the text and show results in Figure 6. Both tendencies of unit language and text are consistent, suggesting the unit language could play the role of text. This also confirms the effect of CL and CM is distinct and explain why both the two tasks work well.

4.3 Why CM and CL "Conflict"?

Previous work identified a gap between CM and CL in speech-to-text translation (Xu et al., 2021). We examine if this issue also causes inconsistency here with the *localness* metric. The metric sums self-attention weights within a window, and lower localness indicates global attention focus. We randomly sample 200 audio recordings from the training dataset and extract attention weights with a window size of 10 in each layer.

Figure 7 shows the results of different models. The localness of the baseline increases initially and reaches its peak at layers 4 to 6 before decreasing. This indicates that S2ST requires cross-modal processing first, followed by cross-lingual processing. CM works in the middle layer and shows higher localness. CL in the upper layers shows higher locality, which is affected by cross-lingual learning. Considering the conflict of using the two together, our conjecture is that the guidance of CM in the middle layer affects the learning of CL at the top level, resulting in the two tasks being unable to help each other. Therefore, we designed task prompts to alleviate the learning conflict.

4.4 How to Make CM and CL Harmonious?

We show the change in localness after applying the task prompt in Figure 8. Although the overall tendency of the two methods are completely different, it can be observed that the consistent changes come from higher layers. When not using task prompt, there will be a significant concussion on the around 8th layer, which we think is the disturbance caused by CM to CL. After applying prompt modeling, CL is easier to learn the semantic information, thus localness is lower at the top level and more inclined to learn global information. Thus designing a strat-

Models	En-Es			En-Fr		
	r=0	r=2	r=4	r=0	r=2	r=4
Baseline	23.0	23.0	23.0	18.8	18.8	18.8
+ CM	21.7	23.5	22.3	17.9	19.5	19.0
+ CM&CL	23.1	**23.8**	23.8	19.6	**20.4**	19.8

Table 4: The performance of different T-Enc layers in applying the CL training.

Models	En-Es			En-Fr		
	K=2	K=3	K=4	K=2	K=3	K=4
Baseline	23.0	23.0	23.0	18.8	18.8	18.8
+ CM	23.3	23.5	23.7	19.5	19.5	19.5
+ CL	23.1	23.9	**24.0**	**19.9**	**19.9**	19.8

Table 5: The performance of different values of K in generating the unit language.

Models	Es-En	Fr-En	En-Es	En-Fr	Avg.
Baseline	19.1	20.3	23.0	18.8	20.3
+ BPE	19.6	20.3	23.0	19.8	20.7 (+0.4)
+ Unit language	**19.7**	**21.0**	**23.8**	**20.4**	**21.2** (+0.9)

Table 6: Comparison of BPE method (Shen et al., 2024) and unit language.

egy to avoid the conflict is necessary.

4.5 Effect of Hyper Parameters

There are two hyperparameters here, r and K. We first test adding the $\mathcal{L}_{\text{CM}}$ at three different layers, and the results are shown in Table 4. If the CM training is applied in the same layer as source unit training, the two tasks conflict and hurt the performance significantly. This proves the effect of unit language differs greatly from that of the unit. Additionally, if the CM training is applied near the CL training, the performance degrades. This also confirms the previous conclusion that there is a conflict between CL and CM training.

Table 5 shows the results for different values of K. K represents the maximum size used to build the unit language. We find that K is highly related to the language pairs, as each language has its unique pronunciation units. Furthermore, there is a threshold for each language pair, meaning that increasing the size of K beyond this threshold does not improve performance. This confirms that unit language is highly related to the language features.

4.6 Comparison with BPE Method

Previous work (Shen et al., 2024) used the BPE method to generate a pseudo language. We reproduced this method based on our multi-task training and normalized units to compared with the proposed unit langauge. We found that the unit language significantly outperforms the BPE method, as shown in Table 6. This is because the BPE method does not consider phrase information, while our method applies n-gram modeling, which makes the unit language more accurate.

5 Related Work

Jia et al. (2019) successfully built S2ST with auxiliary text tasks. Another approach to constructing direct S2ST is by combining advanced end-to-end speech translation methods with unit-based text-to-speech methods (Inaguma et al., 2023; Barrault et al., 2023). Due to the scarcity of training data, some studies utilize unsupervised methods or data augmentation to enhance performance (Jia et al., 2022; Dong et al., 2022; Popuri et al., 2022). A challenge in textless S2ST is extracting acoustic and semantic features from noisy speech sequences, leading many studies to employ the VQ-VAE method to aid alignment learning between different language speeches (Tjandra et al., 2019; Zhang et al., 2020). Conversely, Lee et al. (2022a,b); Chen et al. (2023) regard unit tokens as language text. Our analysis aims to further explore the next steps in unit-based S2ST. Some studies focus on the voice, style, and speed of speech synthesis (Jia et al., 2021; Song et al., 2023; Huang et al., 2022; Fang et al., 2023), while our goal is to enhance the modeling ability of S2ST. Related works generate pseudo language based on the byte-pair encoding method (Wu et al., 2023; Shen et al., 2024). Their work focuses on speech-to-text or text-to-speech tasks, while our work aims to improve the more complex textless speech-to-speech task. Furthermore, their method does not consider in-context information when building the language.

6 Conclusion

Textless S2ST has attracted significant attention from researchers, yet it encounters cross-modal and cross-lingual challenges that impede performance improvement. We introduce unit language to boost either cross-modal or cross-lingual modeling of S2ST. Our analyses demonstrate that CM enhances speech modeling, while CL enhances semantic understanding. We further propose task prompt learning to mitigate conflicts between CM and CL training. Our method achieves comparable performance of textless S2ST to text-based models.

Acknowledgement

This work was supported in part by the National Science Foundation of China (Nos. 62276056 and U24A20334), the Natural Science Foundation of Liaoning Province of China (2022-KF-26-01), the Fundamental Research Funds for the Central Universities (Nos. N2216016 and N2316002), the Yunnan Fundamental Research Projects (No. 202401BC070021), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No.B16009). The authors would like to thank anonymous reviewers for their comments.

Limitations

Our work discusses the modeling challenge we are facing in textless speech-to-speech translation and indicates the conflict between CM and CL. However, the investigation of the proper method to solve the conflict is limited. We introduce the task prompt that could mitigate this conflict with source and target text, but the conflict is not completely cleared. Additionally, while we focus on the translation performance, our work lacks human evaluation which could test the tone and fluency.

References

PD Aguero, Jordi Adell, and Antonio Bonafonte. 2006. Prosody generation for speech-to-speech translation. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages I–I. IEEE.

Loïc Barrault, Yu-An Chung, Mariano Cora Meglioli, David Dale, Ning Dong, Paul-Ambroise Duquenne, Hady Elsahar, Hongyu Gong, Kevin Heffernan, John Hoffman, et al. 2023. Seamlessm4t-massively multilingual & multimodal machine translation. *arXiv preprint arXiv:2308.11596*.

Francisco Casacuberta, Hermann Ney, Franz Josef Och, Enrique Vidal, Juan Miguel Vilar, Sergio Barrachina, Ismael Garcıa-Varea, David Llorens, César Martınez, Sirko Molau, et al. 2004. Some approaches to statistical and finite-state speech-to-speech translation. *Computer Speech & Language*, 18(1):25–47.

William Chan, Navdeep Jaitly, Quoc V Le, and Oriol Vinyals. 2015. Listen, attend and spell. *arXiv preprint arXiv:1508.01211*.

Peng-Jen Chen, Kevin Tran, Yilin Yang, Jingfei Du, Justine Kao, Yu-An Chung, Paden Tomasello, Paul-Ambroise Duquenne, Holger Schwenk, Hongyu Gong, Hirofumi Inaguma, Sravya Popuri, Changhan Wang, Juan Pino, Wei-Ning Hsu, and Ann Lee. 2023. Speech-to-speech translation for a real-world unwritten language. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4969–4983, Toronto, Canada. Association for Computational Linguistics.

Qianqian Dong, Fengpeng Yue, Tom Ko, Mingxuan Wang, Qibing Bai, and Yu Zhang. 2022. Leveraging Pseudo-labeled Data to Improve Direct Speech-to-Speech Translation. In *Proc. Interspeech 2022*, pages 1781–1785.

Qingkai Fang, Yan Zhou, and Yangzhou Feng. 2023. Daspeech: Directed acyclic transformer for fast and high-quality speech-to-speech translation. *ArXiv*, abs/2310.07403.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 369–376, New York, NY, USA. Association for Computing Machinery.

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.

Rongjie Huang, Jinglin Liu, Huadai Liu, Yi Ren, Lichao Zhang, Jinzheng He, and Zhou Zhao. 2022. Transpeech: Speech-to-speech translation with bilateral perturbation. In *The Eleventh International Conference on Learning Representations*.

Hirofumi Inaguma, Sravya Popuri, Ilia Kulikov, Peng-Jen Chen, Changhan Wang, Yu-An Chung, Yun Tang, Ann Lee, Shinji Watanabe, and Juan Pino. 2023. UnitY: Two-pass direct speech-to-speech translation with discrete units. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15655–15680, Toronto, Canada. Association for Computational Linguistics.

Ye Jia, Yifan Ding, Ankur Bapna, Colin Cherry, Yu Zhang, Alexis Conneau, and Nobu Morioka. 2022. Leveraging unsupervised and weakly-supervised data to improve direct speech-to-speech translation. In *Proc. Interspeech 2022*, pages 1721–1725.

Ye Jia, Michelle Tadmor Ramanovich, Tal Remez, and Roi Pomerantz. 2021. Translatotron 2: High-quality direct speech-to-speech translation with voice preservation. In *International Conference on Machine Learning*.

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. In *Proc. Interspeech 2019*, pages 1123–1127.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, Juan Pino, and Wei-Ning Hsu. 2022a. Direct speech-to-speech translation with discrete units. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3327–3339, Dublin, Ireland. Association for Computational Linguistics.

Ann Lee, Hongyu Gong, Paul-Ambroise Duquenne, Holger Schwenk, Peng-Jen Chen, Changhan Wang, Sravya Popuri, Yossi Adi, Juan Pino, Jiatao Gu, and Wei-Ning Hsu. 2022b. Textless speech-to-speech translation on real data. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 860–872, Seattle, United States. Association for Computational Linguistics.

Xinjian Li, Ye Jia, and Chung-Cheng Chiu. 2023. Textless direct speech-to-speech translation with discrete speech representation. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Adam Polyak, Yossi Adi, Jade Copet, Eugene Kharitonov, Kushal Lakhotia, Wei-Ning Hsu, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. Speech resynthesis from discrete disentangled self-supervised representations. *arXiv preprint arXiv:2104.00355*.

Sravya Popuri, Peng-Jen Chen, Changhan Wang, Juan Pino, Yossi Adi, Jiatao Gu, Wei-Ning Hsu, and Ann Lee. 2022. Enhanced Direct Speech-to-Speech Translation Using Self-supervised Pre-training and Data Augmentation. In *Proc. Interspeech 2022*, pages 5195–5199.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Feiyu Shen, Yiwei Guo, Chenpeng Du, Xie Chen, and Kai Yu. 2024. Acoustic bpe for speech generation with discrete tokens. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11746–11750. IEEE.

Kun Song, Yi Ren, Yi Lei, Chunfeng Wang, Kun Wei, Lei Xie, Xiang Yin, and Zejun Ma. 2023. StyleS2ST: Zero-shot Style Transfer for Direct Speech-to-speech Translation. In *Proc. INTERSPEECH 2023*, pages 42–46.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Enrique Vidal. 1997. Finite-state speech-to-speech translation. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 111–114. IEEE.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 993–1003, Online. Association for Computational Linguistics.

Felix Wu, Kwangyoun Kim, Shinji Watanabe, Kyu J Han, Ryan McDonald, Kilian Q Weinberger, and Yoav Artzi. 2023. Wav2seq: Pre-training speech-to-text encoder-decoder models using pseudo languages. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Chen Xu, Bojie Hu, Yanyang Li, Yuhao Zhang, Shen Huang, Qi Ju, Tong Xiao, and Jingbo Zhu. 2021. Stacked acoustic-and-textual encoding: Integrating the pre-trained models into speech translation encoders. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2619–2630, Online. Association for Computational Linguistics.

Chen Zhang, Xu Tan, Yi Ren, Tao Qin, Kejun Zhang, and Tie-Yan Liu. 2020. Uwspeech: Speech to speech translation for unwritten languages. In *AAAI 2021*.

Yuhao Zhang, Chen Xu, Bei Li, Hao Chen, Tong Xiao, Chunliang Zhang, and Jingbo Zhu. 2023. Rethinking and improving multi-task learning for end-to-end speech translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10753–10765, Singapore. Association for Computational Linguistics.

Appendix

A Data Details

We have carried out experiments on four languages of Voxpupil, and the specific data of each language is shown in the Table 7.

Language	Hours(h)	Sentence(k)
Es-En	530	158
Fr-En	521	155
En-Es	413	125
En-Fr	444	135

Table 7: Training data size of the VoxPopuli 4 languages.

B Hyper Parameters of Language Modeling

We primarily apply a 2-gram model here though our method can easily be extended to higher n-gram models. This is because that the computation process becomes very complex and time-consuming with higher n-grams. Table 8 shows the time consumption with the increase of K for one language. Higher values of K require significantly longer computation times to estimate the probabilities accurately. We found that $K = 3$ is sufficient for most languages according to the previous analysis. In summary, the current selection of n-gram and K to merge units is based on a balance of efficiency and empirical study.

K	Time
2	$\sim$2 hours
3	$\sim$12 hours
4 (with pruning strategy)	$\sim$2 days

Table 8: Time consuming of producing the unit language. The size of one corpus is about 160 million units.

C Necessity of Additional Decoders

Another alternative way to use the unit language is with CTC (Graves et al., 2006) to predict the objective, which can avoid the need for additional decoders. We compare the two methods, namely CTC and cross-entropy (CE) based on additional decoders, in Table 9. We found that the additional decoders stabilize the whole modeling process, while CTC fails to take advantage of the unit language.

	CTC loss	Decoder $w/$ CE loss
Baseline	23.0	23.0
+Src unit langauge	22.8(-0.2)	23.9(+0.9)
+Tgt unit langauge	17.3(-5.7)	23.9(+0.9)

Table 9: Using CTC loss or CE loss to leverage the unit language on En-Es task.

D Task Prompt Based on CM$^{'}$ and CL$^{'}$

We test the task prompt training based on $\mathcal{L}_{\text{CM}'}$ and $\mathcal{L}_{\text{CL}'}$. Results shown in Table 10 show that the task prompt still works well and the conflict between CL and CM is not caused by the unit language.

E Sparseness Results on Spanish

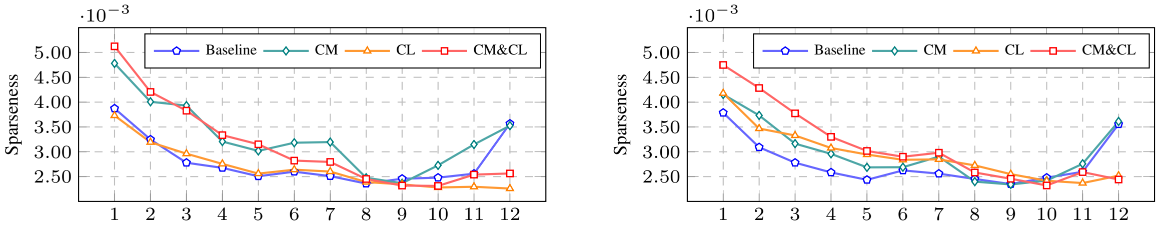
Due to the page limit, we show the effect of the model in Spanish after using CL and CM respectively in the Figure 9. It can be seen that the conclusion is consistent with the French phenomenon described in the main content. We also exhibit Sparseness results with text in Figure 10. The phenomena are the same as those in the previous experiments.

F Localness Results Based on CM$^{'}$ and CL$^{'}$

The localness results of models which apply the CM$^{'}$ and CL$^{'}$ are shown in Figure 11. The trend shown in the figure is consistent with the use of CL and CM methods described in the text, demonstrating that both methods achieve the goals of cross-lingual learning and cross-modal learning.

G Effect of Norm Unit

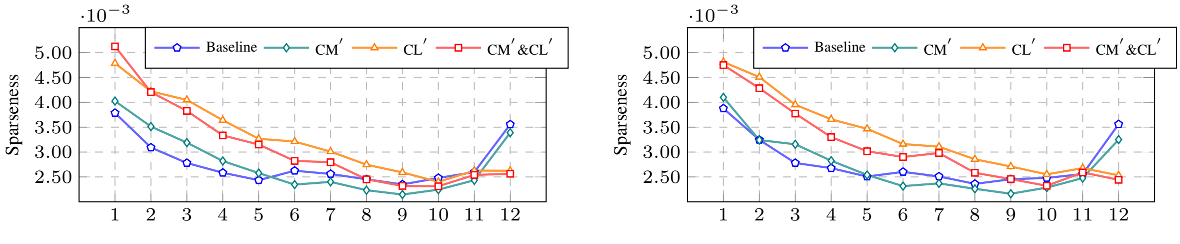
The units used in this work have been normalized Lee et al. (2022b) to reduce noise and other unimportant information. This baseline method in our work achieves about a 3 BLEU improvement, highlighting the importance of unit normalization. If the units are not normalized, unit language does not perform well as shown in Table 11.



(a) Es-En

(b) En-Es

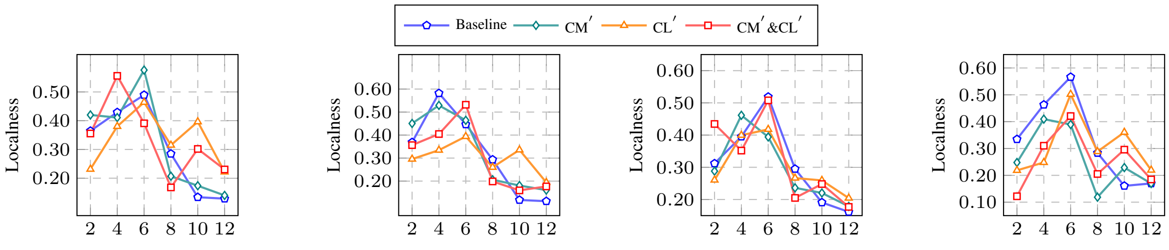
Figure 9: The influence of CM and CL on Es-En (left) and En-Es (right) tasks.



(a) Es-En

(b) En-Es

Figure 10: The influence of $\mathrm{CM}^{'}$ and $\mathrm{CL}^{'}$ on Es-En (left) and En-Es (right) tasks.



(a) Es-En

(b) Fr-En

(c) En-Es

(d) En-Fr

Figure 11: Localness of attention weight on different tasks.

Models	Es-En	Fr-En	En-Es	En-Fr	Avg.
Baseline	19.1	20.3	23.0	18.8	20.3
+ $\mathcal{L}_{\mathrm{CM}'}$&$\mathcal{L}_{\mathrm{CL}'}$	19.8 (+0.7)	20.8 (+0.5)	24.0 (+1.0)	20.8 (+2.0)	21.4 (+1.1)
+Task prompt	**20.3** (+1.2)	**21.4** (+1.1)	**24.6** (+1.6)	**20.9** (+2.1)	**21.8** (+1.5)

Table 10: Performance on different datasets.

Models	Es-En	Fr-En	En-Es	En-Fr	Avg.
Baseline	19.1	20.3	23.0	18.8	20.3
+ $\mathcal{L}_{\mathrm{CM}}$&$\mathcal{L}_{\mathrm{CL}}$ w/o norm unit	19.3	20.3	22.9	19.3	20.5 (+0.2)
+ $\mathcal{L}_{\mathrm{CM}}$&$\mathcal{L}_{\mathrm{CL}}$	**19.7**	**21.0**	**23.8**	**20.4**	**21.2** (+0.9)

Table 11: Comparison of unit language based on norm and un-norm unit.