UTRo-NAST: Non-Autoregressive Speech Translation via Understanding, Translation, and Reordering

Yu-Chen Kuan and Kuan-Yu Chen
National Taiwan University of Science and Technology, Taiwan
E-mail: yckuan@nlp.csie.ntust.edu.tw; kychen@mail.ntust.edu.tw

***Abstract*—Non-autoregressive speech translation (NAR-ST) has attracted increasing attention due to its ability to deliver competitive translation quality with faster inference speed. However, further improving the performance of NAR-ST models remains a challenging and important research direction. To this end, we propose UTRo-NAST, a novel NAR-ST framework that decomposes speech translation into three consecutive subtasks: source speech understanding, word-level translation, and target-side word reordering. The model first encodes the source speech into high-level acoustic representations, which are then translated into a sequence of target words. Finally, the sequence is reordered to match the syntactic and grammatical structure of the target language. By modularizing the speech translation pipeline, UTRo-NAST enhances model interpretability, robustness, and overall performance. Experiments on the MuST-C benchmark across eight language pairs show that UTRo-NAST consistently outperforms existing NAR-ST models and achieves translation quality comparable to strong autoregressive baselines, while maintaining faster decoding speed. These results demonstrate the effectiveness and scalability of the proposed framework for practical speech translation.**

I. Introduction

With the increasing globalization of social communication, the demand for speech translation (ST) has surged. Traditional ST systems rely on cascaded architectures that sequentially combine automatic speech recognition and machine translation, but suffer from error propagation and high latency. End-to-end speech translation (E2E-ST) has emerged as a more efficient alternative, eliminating intermediate steps to reduce latency and improve overall performance [1]–[4]. E2E-ST models typically follow either an autoregressive (AR) or non-autoregressive (NAR) decoding paradigm. While AR models offer strong translation quality by generating tokens sequentially [5]–[9], they incur high inference latency. NAR models enable parallel decoding and significantly reduce latency, making them well-suited for real-time applications. However, differences in word order or syntax across languages can lead to lower-quality translations, especially for language pairs with substantial structural divergence.

Recent advances in NAR E2E-ST have primarily focused on two decoding paradigms: connectionist temporal classification (CTC) [10] and conditional masked language modeling (CMLM) [11]. CTC-based methods directly generate output sequences from speech without requiring pre-segmentation or explicit length prediction [12]. By introducing a special blank symbol, CTC enables constant-time, fully parallel decoding. However, its core limitation is the conditional independence assumption, which prevents the model from capturing contextual dependencies during inference, thereby reducing translation quality. In contrast, CMLM-based methods adopt an iterative refinement strategy, typically implemented through the mask-predict algorithm [13], [14]. These models repeatedly mask and regenerate low-confidence tokens to gradually improve output quality. While CMLM alleviates the independence assumption inherent in CTC, it depends on an auxiliary length predictor and requires multiple decoding iterations, resulting in slower inference compared to single-pass NAR approaches.

We argue that training a fully integrated NAR E2E-ST model can be inherently complex, difficult to debug, and challenging to optimize due to the tight coupling between components [15], [16]. Motivated by the divide-and-conquer philosophy, we propose decomposing the speech translation process into simpler, more manageable subtasks to enhance both model design and translation performance. To this end, we present UTRo-NAST, a framework for non-autoregressive speech translation composed of an acoustic encoder, a word-by-word mapping encoder, and a reordering decoder, which correspond to three subtasks: source speech understanding, word-level translation, and word reordering in the target language. Each module is optimized with task-specific objectives within a multi-task learning framework, enabling the model to transform source speech into coherent target-language output progressively.

We evaluate UTRo-NAST on the MuST-C corpus [17], covering eight language pairs. Experimental results demonstrate that UTRo-NAST consistently outperforms existing NAR-ST models, achieving an average BLEU improvement of +0.4 to +1.0 over the strongest baseline. Furthermore, it delivers up to 4.74× faster inference compared to representative autoregressive systems, highlighting its effectiveness and efficiency for practical speech translation applications.

II. Proposed Methodology

In this study, we propose a novel framework called **UTRo-NAST**, short for **U**nderstanding, **T**ranslation, and **Reo**rdering

979-8-3315-7206-8/25/$31.00

for **N**on-**A**utoregressive **S**peech **T**ranslation. The core idea of UTRo-NAST is to decompose the end-to-end speech translation (E2E-ST) task into three explicitly defined, consecutive subtasks: source speech understanding, word-level translation, and target-side word reordering. Accordingly, UTRo-NAST is composed of three specialized components—an acoustic encoder, a word-by-word mapping encoder, and a reordering decoder. By adopting a modular design with a multi-task learning strategy, UTRo-NAST improves interpretability, enhances modeling flexibility, and boosts overall translation performance, while maintaining the inference efficiency characteristic of NAR models. The complete architecture is illustrated in Figure 1.

A. Acoustic Encoder

Given a speech translation dataset, each training example is represented as a triplet (x, y, z), where $x = (x_1, \ldots, x_T)$ denotes the input acoustic feature sequence (e.g., log-Mel filterbanks), $y = (y_1, \ldots, y_S)$ is the corresponding source-language transcription, and $z = (z_1, \ldots, z_U)$ is the translation in the target language.

To capture both local temporal patterns and global dependencies in the input acoustic feature sequence x, we adopt a stack of Conformer layers to build the acoustic encoder [18]:

$$h^{(l)} = \text{Conformer}^{(l)}(h^{(l-1)}), \quad \text{for } l = 1, \ldots, L, \tag{1}$$

where $h^{(l)}$ denotes the hidden representation at the l-th Conformer layer, and $h^{(0)} = x$. The final output $h^{(L)}$ is used as the input for the downstream translation module.

To explicitly preserve the semantic content of the source speech, we employ the source-language transcription y as an auxiliary supervision signal, using Connectionist Temporal Classification (CTC) loss as the training objective of the acoustic encoder:

$$\mathcal{L}_{\text{AE}} = -\log P_{\text{CTC}}(y \mid h^{(L)}). \tag{2}$$

In addition, we incorporate two auxiliary techniques, Intermediate CTC (InterCTC) [19] and Prediction-Aware Encoding (PAE) [12], [20], to mitigate the conditional independence assumption in CTC and improve training stability at selected intermediate layers. InterCTC facilitates better gradient flow and encourages transcription-aware representations. The InterCTC loss at the l-th layer is formulated as:

$$\mathcal{L}^{(l)}_{\text{InterCTC}} = -\log P_{\text{CTC}}(y \mid h^{(l)}). \tag{3}$$

In parallel, we apply PAE to inject token-level prediction feedback into the intermediate features. Specifically, a softmax distribution $P_{\text{CTC}}(h^{(l)})$ is computed, projected back to the hidden space via a linear transformation, and added to the original representation:

$$\tilde{h}^{(l)} = h^{(l)} + \text{Linear}(P_{\text{CTC}}(h^{(l)})), \tag{4}$$

where $\text{Linear}(\cdot)$ maps the token distribution to the hidden dimension. The resulting feedback-enhanced representation $\tilde{h}^{(l)}$ is then forwarded to the subsequent Conformer block.

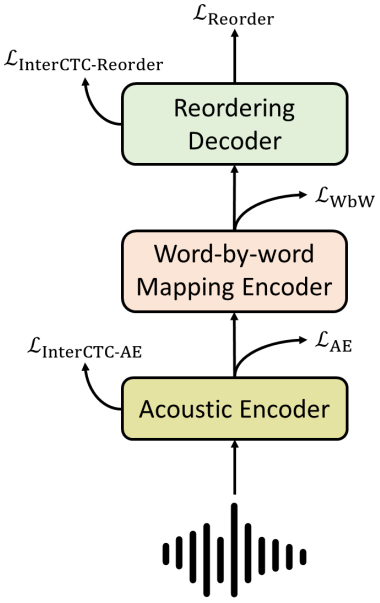

Fig. 1. Overall architecture of the proposed UTRo-NAST framework.

B. Word-by-Word Mapping Encoder

After extracting the acoustic features, the next step is to translate the source speech content into the target language. To simplify this process, we design this stage to perform translation in a monotonic, word-by-word manner.

To bridge the modality gap between acoustic and textual representations, we first apply a linear projection to map the acoustic embeddings $h^{(L)}$ into a latent space aligned with token embeddings:

$$\hat{h} = \text{Linear}(h^{(L)}). \tag{5}$$

The projected sequence $\hat{h}$ is then passed through a stack of sequence modeling blocks (e.g., Transformer or Conformer):

$$\bar{h}^{(k)} = f^{(k)}(\bar{h}^{(k-1)}), \quad \text{for } k = 1, \ldots, K, \tag{6}$$

where $f^{(k)}$ denotes the k-th sequence modeling layer, and $\bar{h}^{(0)} = \hat{h}$.

To achieve word-by-word translation, the encoder is trained to produce a word sequence z' in the target language, where z' is a rough word-level translation that preserves the word order of the source sentence (see Section III-A and Table I for details). While z' may not conform to the correct target-language syntax, it serves as a useful intermediate objective that encourages the model to learn a monotonic word-level translation mapping. The training objective is defined using a CTC loss:

$$\mathcal{L}_{\text{WbW}} = -\log P_{\text{CTC}}(z' \mid \bar{h}^{(K)}), \tag{7}$$

where $\bar{h}^{(K)}$ denotes the final representation output by the sequence modeling stack.

Source Text	I	am	going	to	talk	today	about	energy	and	climate
Word-by-word Mapping	Ich	bin	gehen	Zu	sprechen	Heute	um	Energie	Und	Klima
Final Translation	Heute spreche ich zu Ihnen über Energie und Klima.									

TABLE I
AN EXAMPLE ILLUSTRATING THE WORD-BY-WORD MAPPING AND THE FINAL REORDERED TRANSLATION IN THE UTRO-NAST FRAMEWORK.

C. Reordering Decoder

Building on the preliminary word-by-word translation, the final stage aims to reorder the intermediate word-level sequence into a fluent and grammatically correct sentence in the target language. To achieve this, the reordering decoder takes $\bar{h}^{(K)}$ as input and applies a stack of decoder layers:

$$\tilde{h}^{(j)} = g^{(j)}(\tilde{h}^{(j-1)}, \bar{h}^{(K)}), \quad \text{for } j = 1, \ldots, J, \tag{8}$$

where $g^{(j)}$ denotes the j-th decoder block and $\tilde{h}^{(0)} = \bar{h}^{(K)}$. Each decoder layer employs a cross-attention mechanism, using $\tilde{h}^{(j-1)}$ as the query and $\bar{h}^{(K)}$ as the key and value, to guide the reordering process with global semantic context.

The decoder is trained using a CTC loss based on the ground-truth target sequence z:

$$\mathcal{L}_{\text{Reorder}} = -\log P_{\text{CTC}}(z \mid \tilde{h}^{(J)}). \tag{9}$$

As in earlier stages, InterCTC and PAE are applied to selected intermediate decoder layers to inject semantic supervision and enhance reordering quality.

D. The UTRo-NAST Framework

In summary, UTRo-NAST decomposes the end-to-end speech translation (E2E-ST) task into three consecutive subtasks: source speech understanding, word-level translation, and target-side word reordering. Each subtask is handled by a dedicated module with its own objective, and the entire framework is trained end-to-end using a multi-task loss:

$$\begin{aligned}\mathcal{L} = {} & \alpha_{\text{AE}} \cdot \mathcal{L}_{\text{AE}} + \alpha_{\text{WbW}} \cdot \mathcal{L}_{\text{WbW}} + \alpha_{\text{Reorder}} \cdot \mathcal{L}_{\text{Reorder}} \\ & + \beta_{\text{AE}} \cdot \left(\frac{1}{M} \sum_{m=1}^{M} \mathcal{L}^{(m)}_{\text{InterCTC-AE}} \right) \\ & + \beta_{\text{Reorder}} \cdot \left(\frac{1}{N} \sum_{n=1}^{N} \mathcal{L}^{(n)}_{\text{InterCTC-Reorder}} \right),\end{aligned} \tag{10}$$

where $\mathcal{L}^{(m)}_{\text{InterCTC-AE}}$ denotes the auxiliary InterCTC loss computed at the m-th selected intermediate layer in the acoustic encoder, with M being the total number of such layers. Similarly, $\mathcal{L}^{(n)}_{\text{InterCTC-Reorder}}$ and N represent the corresponding InterCTC losses and number of selected layers in the reordering decoder. The coefficients α_{AE}, α_{WbW}, and α_{Reorder} control the contributions of the primary loss terms, while β_{AE} and β_{Reorder} weight the auxiliary losses.

During inference, UTRo-NAST performs fully parallel decoding. The final output is generated by the reordering decoder through greedy decoding over the CTC output, followed by the removal of blank symbols and repeated tokens. This enables a fast and efficient non-autoregressive decoding process, while maintaining strong translation quality.

III. EXPERIMENTAL SETUP

A. Dataset and Preprocessing

We conduct our experiments on the MuST-C v1 corpus [17], a multilingual speech translation dataset based on TED talks. It covers eight language pairs from English (En) to German (De), Spanish (Es), French (Fr), Italian (It), Dutch (Nl), Portuguese (Pt), Romanian (Ro), and Russian (Ru). All models are trained in a bilingual setting, with model selection performed on the development set and final results reported on the `tst-COMMON` set for each language pair.

To support the auxiliary training objective of the word-by-word mapping encoder, we construct word-level translation targets using a bilingual dictionary generated via Google Translate. Specifically, we begin by extracting all unique English words from the training corpus and translating each word individually into the target language. Hence, for each English sentence, a word-level translation is then created by substituting each word with its corresponding translation while preserving the original word order. This procedure is applied to all language pairs to ensure consistent construction of supervision data. An illustrative example is provided in Table I.

For data preprocessing, we follow the standard pipeline provided by the Fairseq toolkit [3]. Speech utterances shorter than 5 frames or longer than 3,000 frames are filtered out. Acoustic features are extracted as 80-dimensional Mel filterbanks using a 25 ms window and a 10 ms stride. Word segmentation is performed with SentencePiece, using a shared vocabulary of 10,000 tokens for both source and target languages to facilitate cross-lingual modeling. To improve model robustness, we apply SpecAugment [21], while speed perturbation is not used in our experiments.

B. Training Configuration

Our method is implemented using the Fairseq toolkit [3]. We adopt the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$, following the default learning rate schedule provided by Fairseq. A dropout rate of 0.15 is applied during training.

To reduce design complexity and isolate the effects of architectural variations, we adopt a consistent configuration for the sequence modeling blocks $f(\cdot)$ and $g(\cdot)$ used in the word-by-word mapping encoder and reordering decoder, respectively. Specifically, we explore two variants of our framework: **UTRo-NAST-T**, where both components use Transformer blocks, and **UTRo-NAST-C**, where both use Conformer blocks. This

setup allows for a fair comparison between architectures while avoiding the combinatorial explosion of hybrid configurations.

The acoustic encoder, word-by-word mapping encoder, and reordering decoder are configured with $L = 12$, $K = 3$, and $J = 9$ blocks, respectively. Each block consists of 512 hidden units, 8 attention heads, and a feed-forward dimension of 2048.

To enhance training stability and representation learning, InterCTC and PAE are applied at layers 6 and 9 of the acoustic encoder, and at layers 3 and 6 of the word-by-word mapping encoder. For the reordering decoder, a curriculum learning mixing (CLM) mechanism [12], [22], [23] is integrated into its PAE component with a fixed mixing ratio of 0.8. All loss weighting coefficients are set to 1, i.e., $\alpha_{\mathrm{AE}} = \alpha_{\mathrm{WbW}} = \alpha_{\mathrm{Reorder}} = \beta_{\mathrm{AE}} = \beta_{\mathrm{Reorder}} = 1$.

All models are trained on four NVIDIA RTX 3090 GPUs with a maximum batch size of 20,000 tokens. Gradient accumulation is performed with an update frequency of 4. Training is conducted for up to 100 epochs per language pair, with early stopping based on development set performance and a patience of 20 epochs. During inference, we average the parameters of the top 10 checkpoints selected by development set BLEU scores. Evaluation is performed using case-sensitive SacreBLEU [24].

IV. Experiments

A. Main Results

We begin by evaluating the proposed UTRo-NAST framework under two architectural variants: UTRo-NAST-T and UTRo-NAST-C. In addition, we report results with and without sequence-level knowledge distillation (Seq-KD) [40], where distilled translations are generated using an autoregressive machine translation model with a beam size of 5. The results are summarized in the bottom section of Table II.

Overall, UTRo-NAST-C consistently outperforms UTRo-NAST-T in the without Seq-KD setting, achieving an average BLEU score of 29.2 compared to 28.6. This performance gap is particularly evident in syntactically flexible languages, such as Romanian (25.1 vs. 24.1) and Russian (18.6 vs. 17.5), suggesting that the Conformer-based model is better equipped to capture long-range semantic and structural dependencies. When Seq-KD is applied, both variants show consistent improvements. UTRo-NAST-C + Seq-KD achieves an average BLEU score of 29.7, while UTRo-NAST-T + Seq-KD reaches 29.6. The narrowed performance gap indicates that knowledge distillation helps mitigate architectural differences by providing smoother and more deterministic training targets.

B. Comparison with Representative NAR-ST Models

We further compare the proposed UTRo-NAST framework with several representative non-autoregressive speech translation (NAR-ST) models, including Orthros-CTC [13], [14], Orthros-CMLM [13], [14], CTC-NAST [12], Hard Multi-task CTC [27], and NAST-S2T [39]. The comparison results are summarized in the lower section of Table II. Overall, UTRo-NAST achieves superior translation performance across most language pairs. Among these models, CTC-NAST exhibits the strongest performance. However, our UTRo-NAST-T surpasses CTC-NAST on seven out of eight language pairs, with only a slight drop for Romanian (24.1 vs. 24.7). Furthermore, UTRo-NAST-C consistently outperforms CTC-NAST in all cases, highlighting the effectiveness of Conformer-based modeling in capturing both acoustic and semantic dependencies.

When sequence-level knowledge distillation (Seq-KD) is applied, the performance gap between CTC-NAST and the UTRo-NAST variants narrows. This suggests that the distilled supervision helps guide different architectures toward a similar solution space. Nevertheless, UTRo-NAST-C maintains a consistent advantage, demonstrating its architectural robustness even under strong supervision signals. In summary, the UTRo-NAST family delivers steady and reliable improvements over existing NAR-ST models without relying on iterative refinement or complex decoding strategies. These results validate its scalability, cross-lingual robustness, and effectiveness as a fully end-to-end speech translation framework.

C. Comparison with Recent AR-ST Models

We also compare UTRo-NAST variants against several recent autoregressive speech translation (AR-ST) models, as shown in the upper section of Table II. AR-ST approaches such as CTC-Aug ST [12], BiL-CTC + Rescoring [26], ZEROSWOT-MEDIUM [28], and CAST [37] generally demonstrate strong performance, with average BLEU scores ranging from 29 to 31. However, we also observe substantial variance across AR-ST systems, with reported scores ranging from 25 to 31. A closer inspection reveals that many AR-ST models incorporate complex rescoring mechanisms or rely on large pretrained foundation models to enhance translation quality. In contrast, UTRo-NAST is trained in a purely end-to-end manner, without such external augmentation. Despite this, it outperforms several AR-ST baselines, including FCCL (Medium) [25], CMOT [8], and RoPE-ST+Joint [32], even without the use of sequence-level knowledge distillation. Moreover, when Seq-KD is applied, UTRo-NAST exhibits substantial gains and achieves performance on par with several stronger AR-ST baselines. Nevertheless, compared with the top-performing AR-ST models such as ZEROSWOT-MEDIUM, a performance gap remains, indicating that non-autoregressive models still lag behind in terms of ultimate translation accuracy.

D. Inference Speed Analysis

To assess decoding efficiency, we compare the inference speed of UTRo-NAST against both autoregressive (AR-ST) and non-autoregressive (NAR-ST) baselines. Decoding speed is measured on the En-De subset using an NVIDIA RTX 3090 GPU with a batch size of 1. As shown in Table II, UTRo-NAST achieves a speedup of over 3.97$\times$ compared to AR-ST models, such as CTC-Aug ST. While UTRo-NAST-C generally yields better translation performance than UTRo-NAST-T, it incurs slightly higher inference latency due to the increased computational complexity of Conformer blocks. This highlights a trade-off between translation accuracy and decoding speed when

Model	Pretrained Model		BLEU (↑)									Speedup	Speedup*
	Speech	Text	De	Es	Fr	It	Nl	Pt	Ro	Ru	Avg.		
Machine Translation													
Transformer*			31.2	36.0	43.4	32.1	36.2	37.7	30.2	19.9	33.3	-	-
Autoregressive Models													
ESPnet-ST [2]			22.9	28.0	32.7	23.8	27.4	28.0	21.9	15.8	25.1	-	-
Fairseq S2T [3]			22.7	27.2	32.9	22.7	27.3	28.1	21.9	15.3	24.8	-	-
FCCL (Medium) [25]	✓		25.9	30.7	36.8	26.4	30.5	31.8	25.0	17.6	28.1	-	-
CMOT [8]	✓		27.0	31.1	37.3	26.9	31.2	32.7	25.3	17.9	28.7	-	-
CTC-Aug ST [12]			26.9	31.5	38.1	27.4	31.9	33.4	25.8	18.7	29.2	1.0x	-
CTC-Aug ST*			27.0	32.6	38.5	27.9	31.8	32.9	25.3	18.0	29.3	-	1.0x
CTC-Aug ST + Seq-KD [12]			27.7	31.6	39.5	27.5	32.3	33.7	26.6	18.7	29.7	1.0x	-
BiL-CTC + Rescoring [26]			28.4	32.0	39.5	28.2	33.0	34.3	27.0	19.6	30.3	-	-
Hard Multi-Task CTC/Attn [27]	✓		29.2	33.2	39.2	-	-	-	-	-	-	-	-
ZEROSWOT-MEDIUM [28]	✓	✓	30.5	34.9	39.4	30.6	35.0	37.1	27.8	20.3	31.9	-	-
DiG-SST [29]	✓		26.9	30.9	37.6	-	-	-	-	-	-	-	-
ASR + AST SALM [30]	✓	✓	30.7	-	-	-	-	-	-	-	-	-	-
GenTranslate (E2E) [31]	✓	✓	-	33.9	-	29.4	-	-	-	-	-	-	-
RoPE-ST+joint [32]			23.9	28.4	34.2	23.9	28.4	29.5	22.9	16.2	25.9	-	-
S-Align [33]	✓		26.5	31.3	37.6	-	-	-	-	-	-	-	-
Memory-ST [34]			23.2	28.6	33.5	23.9	27.6	28.7	-	-	-	-	-
ST-MKD-MT [35]	✓		28.2	-	39.5	-	32.1	34.1	-	-	-	-	-
TDR [36]	✓	✓	29.8	35.3	41.6	28.5	-	-	-	-	-	-	-
CAST [37]	✓		27.0	32.0	37.6	27.4	31.5	33.1	25.6	18.2	29.1	-	-
PIA-8 [38]			25.5	30.2	35.6	25.1	29.7	30.4	23.0	16.0	26.9	-	-
Non-autoregressive Models													
CTC [13]			24.1	29.0	34.6	24.3	28.5	-	-	-	-	13.83x	-
CMLM [13]		✓	23.5	27.5	33.0	22.7	27.6	-	-	-	-	2.89x	-
Orthros-CTC [14]			25.3	30.4	36.2	25.4	29.9	-	-	-	-	1.14x	-
Orthros-CMLM [14]		✓	24.5	29.1	34.6	24.0	28.5	-	-	-	-	2.73x	-
CTC-NAST [12]			25.8	-	-	-	-	-	-	-	-	5.67x	-
CTC-NAST*			25.9	31.9	37.5	26.5	30.6	31.8	24.7	16.9	28.2	-	4.55x
CTC-NAST + Seq-KD [12]			27.3	31.8	38.9	27.7	32.3	33.3	26.1	18.9	29.5	5.67x	-
Hard Multi-Task CTC [27]	✓		23.4	28.4	33.6	-	-	-	-	-	-	-	-
NAST-S2T [39]			24.5	28.2	-	-	-	-	-	-	-	-	-
UTRo-NAST-T			26.7	32.0	37.8	27.5	31.2	32.0	24.1	17.5	28.6	-	4.74x
UTRo-NAST-T + Seq-KD			27.7	31.9	38.9	27.7	32.4	33.0	26.3	19.0	29.6	-	4.74x
UTRo-NAST-C			26.6	32.7	38.4	27.4	32.2	32.7	25.1	18.6	29.2	-	3.97x
UTRo-NAST-C + Seq-KD			27.3	32.1	38.7	28.0	32.5	33.5	26.3	18.9	29.7	-	3.97x

TABLE II

BLEU SCORES AND DECODING SPEEDS OF VARIOUS SPEECH TRANSLATION MODELS EVALUATED ON THE MUST-C TST-COMMON SET. THE "PRETRAINED MODEL" COLUMNS INDICATE WHETHER ADDITIONAL PRETRAINED SPEECH OR TEXT FOUNDATION MODELS ARE USED. MODELS MARKED WITH "*" DENOTE OUR IMPLEMENTATIONS.

choosing architectural components. In summary, the UTRo-NAST framework offers a strong balance between translation quality and inference efficiency. It consistently delivers faster decoding while achieving performance comparable to that of several autoregressive models across multiple language pairs, demonstrating its effectiveness and scalability for real-world speech translation applications.

V. CONCLUSION

In this work, we propose UTRo-NAST, a novel non-autoregressive speech translation framework that decomposes the complex speech translation process into three simpler and more manageable subtasks. By adopting a divide-and-conquer philosophy, UTRo-NAST achieves competitive translation quality while maintaining high inference efficiency and architectural flexibility. Experimental results on eight language pairs demonstrate the effectiveness of UTRo-NAST: it consistently outperforms existing NAR-ST baselines and achieves translation performance comparable to strong autoregressive models. For future work, we plan to explore the application of UTRo-NAST in low-resource and real-time settings, investigate the integration of large language models to further enhance translation quality, and extend the framework to broader multimodal translation tasks.

ACKNOWLEDGMENT

This work was supported in part by the National Science and Technology Council, Taiwan, under Grant NSTC-112-2628-E-011-008-MY3 and Grant NSTC-114-2640-B-002-005; and in part by the Empower Vocational Education Research

Center, National Taiwan University of Science and Technology, through the Featured Areas Research Center Program under the Higher Education Sprout Project, Ministry of Education, Taiwan. We would like to thank the National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

REFERENCES

[1] R. J. Weiss, J. Chorowski, N. Jaitly, Y. Wu, and Z. Chen, "Sequence-to-sequence models can directly translate foreign speech," in *Proc. of Interspeech*, 2017.

[2] H. Inaguma, S. Kiyono, K. Duh, *et al.*, "ESPnet-ST: All-in-one speech translation toolkit," in *Proc. of ACL*, 2020.

[3] C. Wang, Y. Tang, X. Ma, A. Wu, D. Okhonko, and J. Pino, "Fairseq S2T: Fast speech-to-text modeling with fairseq," in *Proc. of AACL-IJCNLP*, 2020.

[4] C. Zhao, M. Wang, Q. Dong, R. Ye, and L. Li, "NeurST: Neural speech translation toolkit," in *Proc. of ACL-IJCNLP*, 2021.

[5] R. Ye, M. Wang, and L. Li, "End-to-end speech translation via cross-modal progressive training," in *Proc. of Interspeech*, 2021.

[6] Q. Fang, R. Ye, L. Li, Y. Feng, and M. Wang, "STEMM: Self-learning with speech-text manifold mixup for speech translation," in *Proc. of ACL*, 2022.

[7] R. Ye, M. Wang, and L. Li, "Cross-modal contrastive learning for speech translation," in *Proc. of NAACL*, 2022.

[8] Y. Zhou, Q. Fang, and Y. Feng, "CMOT: Cross-modal mixup via optimal transport for speech translation," in *Proc. of ACL*, 2023.

[9] P. Gao, R. Zhang, Z. He, H. Wu, and H. Wang, "An empirical study of consistency regularization for end-to-end speech-to-text translation," in *Proc. of NAACL*, 2024.

[10] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. of ICML*, 2006.

[11] M. G. O. L. Y. Liu and L. Zettlemoyer, "Maskpredict: Parallel decoding of conditional masked language models," in *Proc. of EMNLP*, 2019.

[12] C. Xu, X. Liu, X. Liu, *et al.*, "CTC-based non-autoregressive speech translation," in *Proc. of ACL*, 2023.

[13] H. Inaguma, Y. Higuchi, K. Duh, T. Kawahara, and S. Watanabe, "Non-autoregressive end-to-end speech translation with parallel autoregressive rescoring," *arXiv preprint arXiv:2109.04411*, 2021.

[14] H. Inaguma, Y. Higuchi, K. Duh, T. Kawahara, and S. Watanabe, "Orthros: Non-autoregressive end-to-end speech translation with dual-decoder," in *Proc. of ICASSP*, 2021.

[15] Q. Ran, Y. Lin, P. Li, and J. Zhou, "Guiding non-autoregressive neural machine translation decoding with reordering information," in *Proc. of AAAI*, 2021.

[16] M. Omachi, B. Yan, S. Dalmia, Y. Fujita, and S. Watanabe, "Align, write, re-order: Explainable end-to-end speech translation via operation sequence generation," in *Proc. of ICASSP*, 2023.

[17] M. A. Di Gangi, R. Cattoni, L. Bentivogli, M. Negri, and M. Turchi, "MuST-C: A Multilingual Speech Translation Corpus," in *Proc. of NAACL*, 2019.

[18] A. Gulati, J. Qin, C.-C. Chiu, *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. of Interspeech*, 2020.

[19] J. Lee and S. Watanabe, "Intermediate loss regularization for ctc-based speech recognition," in *Proc. of ICASSP*, 2021.

[20] J. Nozaki and T. Komatsu, "Relaxing the conditional independence assumption of ctc-based asr by conditioning on intermediate predictions," in *Proc. of Interspeech*, 2021.

[21] D. S. Park, W. Chan, Y. Zhang, *et al.*, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proc. of Interspeech*, 2019.

[22] C. Wang, Y. Wu, S. Liu, M. Zhou, and Z. Yang, "Curriculum pre-training for end-to-end speech translation," in *Proc. of ACL*, 2020.

[23] L. Qian, H. Zhou, Y. Bao, *et al.*, "Glancing transformer for non-autoregressive neural machine translation," in *Proc. of ACL-IJCNLP*, 2021.

[24] M. Post, "A call for clarity in reporting BLEU scores," in *Proc. of WMT*, 2018.

[25] H. Zhang, N. Si, Y. Chen, *et al.*, "Improving speech translation by cross-modal multi-grained contrastive learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1075–1086, 2023.

[26] C. Xu, X. Liu, E. He, *et al.*, "Bridging the gaps of both modality and language: Synchronous bilingual ctc for speech translation and speech recognition," in *Proc. of ICASSP*, 2024.

[27] B. Yan, X. Chang, A. Anastasopoulos, Y. Fujita, and S. Watanabe, "Cross-modal multi-tasking for speech-to-text translation via hard parameter sharing," in *Proc. of ICASSP*, 2024.

[28] I. Tsiamas, G. Gállego, J. Fonollosa, and M. Costa-jussà, "Pushing the limits of zero-shot end-to-end speech translation," in *Proc. of ACL*, 2024.

[29] X. Chen, K. Fan, W. Luo, *et al.*, "Divergence-guided simultaneous speech translation," in *Proc. of AAAI*, 2024.

[30] Z. Chen, H. Huang, A. Andrusenko, *et al.*, "Salm: Speech-augmented language model with in-context learning for speech recognition and translation," in *Proc. of ICASSP*, 2024.

[31] Y. Hu, C. Chen, C.-H. Yang, *et al.*, "Gentranslate: Large language models are generative multilingual speech and machine translators," in *Proc. of ACL*, 2024.

[32] X. Li, S. Li, X.-L. Zhang, and S. Rahardja, "Transformer-based end-to-end speech translation with rotary position embedding," *IEEE Signal Processing Letters*, vol. 31, pp. 371–375, 2024.

[33] Y. Zhang, K. Kou, B. Li, *et al.*, "Soft alignment of modality space for end-to-end speech translation," in *Proc. of ICASSP*, 2024.

[34] Y. Yuan, Y. Zhou, and X. Shi, "Memory-augmented speech-to-text translation with multi-scale context translation strategy," in *Proc. of ICASSP*, 2024.

[35] H. Wang, Z. Xue, Y. Lei, and D. Xiong, "End-to-end speech translation with mutual knowledge distillation," in *Proc. of ICASSP*, 2024.

[36] Y. Zhou, Y. Yuan, C. Zhang, and X. Shi, "Boosting context-aware speech translation with large language models," *IEEE Signal Processing Letters*, vol. 32, pp. 1955–1959, 2025.

[37] X. Tian, H. Wei, Z. Gong, J. Li, and J. Xie, "Improving end-to-end speech-to-text translation with document-level context," *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

[38] C. Xu, X. Liu, Y. Zhang, *et al.*, "Unveiling the fundamental obstacle in speech-to-text modeling: Understanding and mitigating the granularity challenge," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 1719–1729, 2025.

[39] Z. Ma, Q. Fang, S. Zhang, S. Guo, Y. Feng, and M. Zhang, "A non-autoregressive generation framework for end-to-end simultaneous speech-to-any translation," in *Proc. of ACL*, 2024.

[40] Y. Kim and A. M. Rush, "Sequence-level knowledge distillation," in *Proc. of EMNLP*, 2016.