CUHKSZ Simultaneous Speech Translation System for IWSLT 2026

Zeyu Yang[1], **Satoshi Nakamura**[1]
[1]The Chinese University of Hong Kong, Shenzhen
Correspondence: zeyuyang1@link.cuhk.edu.cn, snakamura@cuhk.edu.cn

Abstract

We present the **CUHKSZ Team** submission to the IWSLT 2026 Simultaneous Speech Translation evaluation, targeting the main and *Extra Context* tracks for English→{Chinese, German} on unsegmented speech.

Our system is built upon **Qwen3-Omni-30B-A3B**, a natively aligned audio-text LLM. Under the *Constrained* condition, we apply **LoRA** adaptation exclusively to the LLM. Specifically, we construct syntax-aware, chunk-aligned supervision from existing ASR corpora, using **Qwen3-30B-Instruct** to synthesize target translations. This enables the model to internalize the simultaneous read/write policy by autonomously predicting `<wait>` tokens at semantically incomplete boundaries.

With the policy internalized, execution is delegated to a lightweight streaming *agent* served via vLLM. This agent feeds audio in fixed chunks, manages a bounded dialogue history, and enforces strict emission controls to minimize computation-aware delay. For the subtrack, contextual priors are dynamically injected into the prompt.

On the official dev set, our 0–2 s latency regime submissions achieve 40.5 BLEU (1.95 s) for En→Zh and 27.7 BLEU (1.72 s) for En→De. In the 2–4 s regime, performance scales to 42.1 BLEU (2.16 s) and 30.5 BLEU (2.29 s) respectively.

1 Introduction

Simultaneous speech translation (SimulST) requires systems to incrementally generate target-language text while the source speech is ongoing, enabling applications such as live interpretation and broadcast captioning. The IWSLT 2026 Simultaneous Speech Translation shared task (Adelani et al., 2026) pushes this setting in two directions that complicate the deployment of established architectures. First, the test condition is *unsegmented*: systems must process continuous, unbounded audio streams—often lasting tens of minutes—and commit translations dynamically without oracle sentence boundaries. Second, evaluation is governed by the *computation-aware* LongYAAL protocol (Polák et al., 2025), which explicitly incorporates the model's actual inference time into the end-to-end latency measurement. Consequently, a competitive submission must achieve a rigorous balance between translation fidelity and computational efficiency.

Most existing SimulST systems approach this challenge by cascading or tightly coupling separately pre-trained components: an acoustic encoder, a text decoder, and a learned projection module that bridges the two modalities (Seamless Communication et al., 2023; Papi et al., 2023). While effective in offline benchmarks, composite architectures introduce significant overhead in the streaming regime. Processing each audio chunk requires multiple forward passes across disparate models, the projection layer strictly couples the components, and the overall inference footprint complicates adherence to strict latency budgets on standard single-GPU hardware. This architectural complexity is further compounded by an *empirical data bottleneck*: high-quality speech-to-speech translation parallel corpora remain an order of magnitude scarcer than ASR transcripts or text-only data, severely restricting the degree to which large-scale (e.g., 30B-class) decoders can be fully adapted for the SimulST objective.

In this paper we describe the **CUHKSZ Team** submission, which departs from the composite paradigm in two ways. (i) *Architecturally*, we replace the encoder/projection/decoder pipeline with a single *natively aligned* audio-text multimodal LLM, **Qwen3-Omni-30B-A3B** (Qwen Team, 2025a), whose acoustic and textual representations are already fused by large-scale multimodal pre-training; this inherently resolves the

Proceedings of the 23rd International Conference on Spoken Language Translation (IWSLT 2026), pages 111–118

alignment discrepancies typical of joint-training approaches. (ii) *Operationally*, we wrap the model in a lightweight *streaming agent* that maintains a bounded multi-turn conversation with the LLM, feeds each incoming audio window as a new user turn, and applies a strict set of emission controls (minimum-commit length, final-boundary rescue, n-gram repetition avoidance, and a sliding audio context window) to ensure every decision step complies with the computation-aware latency budget. Importantly, the read/write timing policy driving this loop is not hard-coded via heuristics: it is *learned* during training by fine-tuning the backbone on *syntax-aware chunks* (Yang et al., 2026)—units derived from dependency parsing to ensure semantic coherence. This effectively scales the linguistically motivated chunking philosophy from small, dedicated SimulST models to a 30B-class multimodal LLM. Furthermore, this agent generalizes natively to the *Extra Context* sub-track: per-talk named entities (for the low-latency regime) or full paper abstracts (for the high-latency regime) are extracted from the provided PDFs and injected into the system prompt, improving lexical fidelity on scientific vocabulary without altering the streaming logic.

To make this architecture trainable under the official *Constrained with Large Language Models* condition, we address the data bottleneck in two stages. First, we assemble a mixture of LibriSpeech (Panayotov et al., 2015), Common Voice (Ardila et al., 2020), VoxPopuli (Wang et al., 2021a), and CoVoST 2 (Wang et al., 2021b). For the three ASR-only corpora, we synthesize En→Zh and En→De targets using **Qwen3-30B-Instruct** (Qwen Team, 2025b), converting abundant English transcripts into pseudo-parallel SimulST supervision. Second, adapting the syntax-aware methodology (Yang et al., 2026), every training pair is re-segmented into dependency-derived chunks and re-ordered on the target side before being formatted as chunk-aligned training examples. We then adapt the backbone via **LoRA** (Hu et al., 2022) fine-tuning, freezing the audio encoder and updating exclusively the LLM sub-network. The resulting system achieves strong performance across both latency buckets on the official development set: 40.5 BLEU / 73.5 XCOMET-XL at 1.95 s computation-aware latency for En→Zh and 27.7 BLEU / 85.0 XCOMET-XL at 1.72 s for En→De in the 0–2 s bucket, scaling to 42.1 / 75.7 at 2.16 s and 30.5 / 86.6 at 2.29 s respectively in the 2–4 s bucket, effectively covering all main-track language pairs and sub-track operating points.

Contributions.

- A unified streaming SimulST architecture that offloads all simultaneous-decoding heuristics to a *lightweight Python agent*, ensuring rigorous computation-aware latency control while maximizing system reproducibility.
- A *syntax-aware* training methodology that utilizes dependency-derived chunks to supervise the multimodal LLM. This enables the model to autonomously learn read/write timing from linguistic structure, overcoming the limitations of rigid segmentation.
- A LoRA-only adaptation strategy paired with **Qwen3-30B-Instruct**-synthesized targets. This successfully addresses the speech-translation data bottleneck under the "Constrained with LLMs" condition without unfreezing the audio encoder.

The remainder of the paper is organized as follows. Section 2 describes the natively aligned base model and the decoupled system architecture. Section 2 details the streaming agent, including its dialogue scheme and emission controls. Section 3 outlines the training corpora, LLM-based data annotation, and LoRA fine-tuning setup. Appendix B reports the experimental setup and results across both the main and *Extra Context* tracks, and Section 5 concludes our work.

2 System Architecture and Agent

Figure 1 illustrates our end-to-end streaming pipeline. We replace the traditional cascaded architecture with a single, natively aligned audio-text multimodal LLM: **Qwen3-Omni-30B-A3B** (Qwen Team, 2025a). A lightweight Python agent encapsulates the model, acting purely as an executor (pseudocode detailed in Appendix A).

2.1 Atomic Backbone and Context

As shown in Zones 1 and 2 of Figure 1, audio is treated as a first-class input modality. We freeze the audio encoder and adapt only the LLM Thinker via LoRA, disabling the vision and speech-synthesis modules to ensure the 3B-activated MoE model runs efficiently under tight computation-aware latency budgets on a single GPU.

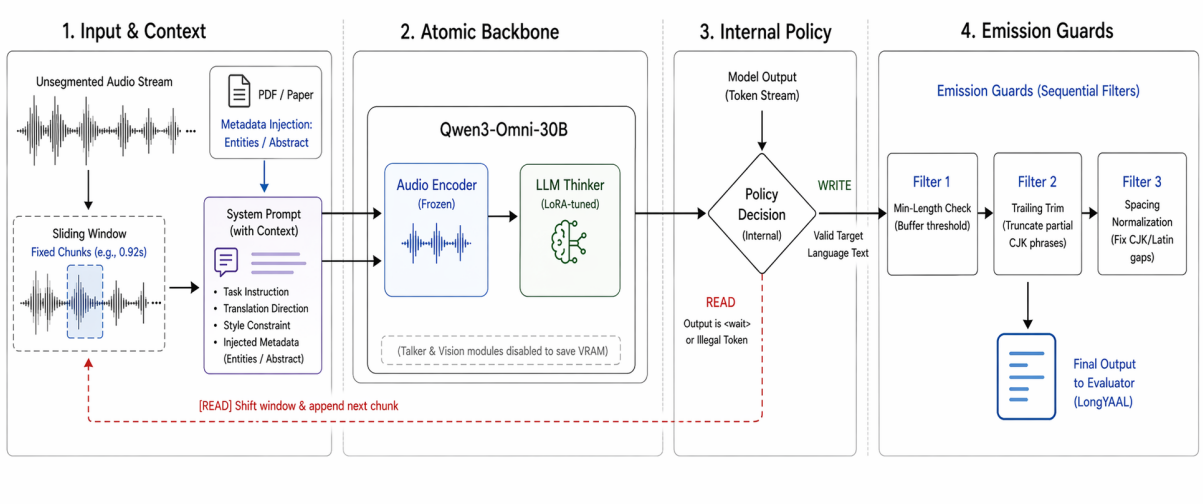

Figure 1: The end-to-end streaming pipeline of the CUHKSZ SimulST system. The process is divided into four zones: context-aware input chunking, the atomic multimodal backbone, the internalized read/write policy, and the sequential emission guards.

The backbone naturally maintains translation history, treating each incoming audio chunk (e.g., 0.92 s) as a new user turn. For the Extra Context sub-track, extracted PDF metadata (named entities or abstracts) is injected directly into the initial system prompt, uniformly aligning with the streaming loop without any architectural changes (Koshkin et al., 2024).

2.2 Internalized Read/Write Policy

Instead of relying on external heuristics, we internalize the read/write policy (Figure 1, Zone 3). The backbone is fine-tuned on syntax-aware chunks (Yang et al., 2026) to emit a dedicated `<wait>` token when the acoustic prefix lacks sufficient translatable information.

During the streaming loop, if the model emits `<wait>`—or if the response is empty, lacks target-language characters, or loops repetitively—the agent intercepts this as a READ action, shifting the window and appending the next audio chunk. Otherwise, it prepares for a WRITE commit.

2.3 Emission Guards

To prevent metric penalization caused by LLM tokenization artifacts, accepted emissions must pass three sequential filtering stages (Figure 1, Zone 4) before reaching the evaluator:

1. **Filter 1 (Min-length Check)**: Ensures the pending buffer reaches a character threshold, avoiding costly single-character emissions under the character-aware LongYAAL penalty.
2. **Filter 2 (Trailing Trim)**: Retains complete linguistic phrases by stripping and rebuffering text that follows sentence-ending CJK punctuation.
3. **Filter 3 (Spacing Normalization)**: Removes spurious spaces at CJK-Latin boundaries, preventing severe artificial XCOMET degradation.

Finally, upon receiving the end-of-source flag, the agent enters a brief drain loop that bypasses the target-language detector, ensuring no acoustic tail is dropped.

3 Data and Training

This section describes how the backbone of Section 2.1 is adapted to emit the read/write policy introduced in Section 2.2. Four design choices shape our training paradigm. *(i)* All supervision data is sourced exclusively from the official IWSLT 2026 *Constrained with Large Language Models* allow-list. *(ii)* Target-language references and the chunk-level read/write alignments that underlie our `<wait>` supervision are produced via a *single unified LLM call per utterance*. This significantly simplifies a data-generation pipeline that previous methodologies typically implemented as three chained specialized models. *(iii)* The resulting data is passed through a multi-dimensional, quality-based filter prior to training. *(iv)* Only a lightweight language-side **LoRA** (Hu et al., 2022) adapter is trained; all other components of the backbone are strictly frozen or disabled.

3.1 Training Corpora

All training audio is sourced exclusively from the official IWSLT 2026 *Constrained with Large Language Models* allow-list. We utilize the English portions of LibriSpeech (Panayotov et al., 2015), Common Voice 17.0 (Ardila et al., 2020), and VoxPopuli (Wang et al., 2021a)—three corpora that collectively encompass read speech, user-contributed single-sentence utterances, and multilingual parliamentary proceedings—as well as CoVoST 2 (Wang et al., 2021b), the only allow-list corpus that provides human-annotated English→German target-language references. CoVoST 2 aside, none of these corpora include translations into our target languages; we synthesize them in-pipeline, as detailed in Section 3.2. Per-corpus statistics and total training example counts are reported in Table 2 (Appendix B).

3.2 Unified LLM-based Annotation

Constructing training data for simultaneous translation is inherently more complex than for offline translation: every *prefix* of the source must be paired with either a safe partial translation or a control token `<wait>`, and both must be aligned to the timeline of the source audio. A conventional pipeline, exemplified by the syntax-aware methodology (Yang et al., 2026), chains three specialized models—spaCy for source-side chunking, Whisper (Radford et al., 2023) for word-level timestamps, and a bilingual word aligner (e.g., SimAlign) for chunk-to-target mapping—to derive `<wait>` decisions heuristically from the intersection of these signals. This approach suffers from compounding errors and disagreements among models that were never explicitly trained to be mutually consistent.

We replace these three specialized models with a single text-only LLM annotator, **Qwen3-32B** (Qwen Team, 2025b), which in one prompted call produces *jointly*: *(a)* the syntax-aware source chunking, *(b)* the bilingual chunk-level alignment to the target language, *(c)* the target-side reordering required when source and target have divergent word orders, and *(d)* the `<wait>` decisions at chunk boundaries. **Qwen3-32B** is served via vLLM and queried via batched inference. Its response is a JSON structure detailing, for each source chunk, the chunk text and either the corresponding partial translation or the literal `<wait>` token. The `<wait>` signal is thus *predicted* by the annotator based on whether the current prefix admits a committable translation under the context seen thus far—a decision that depends on both syntactic completeness and cross-lingual word-order constraints, which is notoriously difficult to derive from rule-based components.

To ensure the resulting chunking remains stable across corpora and compatible with streaming inference, we impose four constraints on every annotator output, all of which are stated in the prompt and verified post-hoc. First, each chunk must contain at most seven source words. Second, the concatenation of all source chunks must equal the original transcript, modulo whitespace. Third, the concatenation of all non-`<wait>` target chunks must match the reference translation. Fourth, the final chunk of any utterance must not be `<wait>`. Subsequently, a single lightweight forced-alignment step using *Whisper-large-v3* (Radford et al., 2023) is applied to the source transcripts to attach start/end timestamps to every chunk; these timestamps constitute the only non-LLM signal utilized in our annotation pipeline.

3.3 Quality Filtering

Raw annotator outputs vary widely in quality, especially on very short or very long utterances. We therefore filter the annotated corpus through a four-dimensional quality score that combines the following components with weights 0.3:0.3:0.3:0.1: *(i)* a *timestamp score* measuring the completeness and monotonicity of the Whisper-provided word-level timestamps; *(ii)* a *text score* combining length, complexity, and sentence-completeness heuristics; *(iii)* an *alignment score* checking that the last timestamp does not overshoot the total audio duration and that the number of timestamps agrees with the number of annotated tokens; and *(iv)* a *decision-balance score* rewarding utterances whose per-utterance READ:WRITE ratio lies near the corpus average of approximately 3:1. Only examples whose aggregate score exceeds a strict retention threshold are kept; the thresholds and the resulting per-language training counts are reported in Table 2.

In addition, a fixed fraction of *edge-case* examples—extremely short or extremely long utterances, questions, multi-chunk sentences, and examples with atypical punctuation—is explicitly preserved during filtering to prevent the final training distribution from becoming too narrow. Edge cases are selected by ranking candidates within each category by their quality score and retaining

Table 1: End-to-end performance on the MCIF development set. Δ denotes the XCOMET change when introducing PDF-derived extra context compared to the *None* baseline. All latency measurements reflect the computation-aware LongYAAL protocol.

Pair	Regime	Target CA Latency	Context Prompt	XCOMET	Δ	BLEU	LongYAAL$_{CA}$
En→Zh	Low	≤ 2000 ms	None (Main Track)	73.54	—	40.46	1954 ms
			Entities (Sub-track)	73.85	+0.31	39.88	1997 ms
	High	≤ 4000 ms	None (Main Track)	75.74	—	42.14	2164 ms
			Abstract (Sub-track)	76.45	+0.71	42.99	2131 ms
En→De	Low	≤ 2000 ms	None (Main Track)	85.04	—	27.72	1721 ms
			Entities (Sub-track)	84.63	−0.41	28.42	1649 ms
	High	≤ 4000 ms	None (Main Track)	86.56	—	30.54	2288 ms
			Abstract (Sub-track)	85.93	−0.63	30.22	2135 ms

the top entries up to a predefined per-category budget.

3.4 LoRA Fine-Tuning

We fine-tune **Qwen3-Omni-30B-A3B** with **LoRA** (Hu et al., 2022) adapters inserted exclusively into the *Thinker* module; every other component of the backbone is either frozen or disabled. Concretely, the audio encoder participates in the forward pass with frozen weights, ensuring that the robust speech-understanding capability acquired during large-scale multimodal pre-training is preserved. The *Talker* (speech-synthesis) and vision components of Qwen3-Omni are completely disabled at training time: our task produces text from audio and never routes information through these modalities, so retaining them in the computation graph would incur unnecessary memory overhead without contributing to the gradient signal. This concentrates all adaptation capacity exactly in the module where the two new behaviors of the system—target-language generation and `<wait>` emission—must be learned.

We train one LoRA checkpoint per target language (English→Chinese and English→German). The two checkpoints share an identical architecture, hyper-parameters, and annotation pipeline; they differ only in the synthetic target-language translations against which they are supervised and in the target-language instruction string of the system prompt (Section 2.1). The complete list of targeted modules, LoRA rank, optimizer configuration, and training budget is reported in Appendix B.

4 Experiments

4.1 Data and Evaluation Protocol

All evaluations are conducted on the official IWSLT 2026 MCIF development set (919 unsegmented utterances) and the ACL-TALKS blind test set. Translation quality is measured using **XCOMET-XL** (Guerreiro et al., 2023) and sacre**BLEU**, computed via OmniSTEval (Polák et al., 2025) after resegmentation. Latency is quantified using the computation-aware (CA) **LongYAAL** protocol, which strictly penalizes hardware decoding time. We target the official *Low* (0–2 s) and *High* (2–4 s) latency regimes using a single NVIDIA A800 GPU.

Our streaming supervision is sourced from LibriSpeech, Common Voice 17.0, CoVoST2, and VoxPopuli. After the 90/100 quality filter (§3.3), we retain 220 K chunk-level training examples for En→Zh and En→De. At inference, the fine-tuned backbone is served via vLLM. The agent uses greedy decoding and prunes the context window to the most recent 16 turns or 20 s of audio. The only latency-governing parameter is `chunk_sec`: 0.92 s / 1.60 s for low-latency En→Zh / En→De, and 1.28 s / 2.56 s for high-latency.

4.2 Main Track and Extra Context Results

Table 1 summarizes our primary submissions. In the main track, the system comfortably satisfies the strict latency ceilings while delivering robust quality. Moving from the low- to high-latency regime yields a consistent +2.2 XCOMET gain on En→Zh and +1.5 on En→De, confirming that the agent exploits the relaxed budget predictably without over-emitting.

For the *Extra Context* sub-track, we inject PDF-derived context directly into the system prompt: heuristic named-entities for low-latency (∼40 tokens) and full abstracts for high-latency (∼300 tokens). On En→Zh, this zero-shot injection consistently improves lexical fidelity, peaking at a +0.71 XCOMET gain for high-latency. On En→De, the quality deltas are slightly negative. Notably, how-

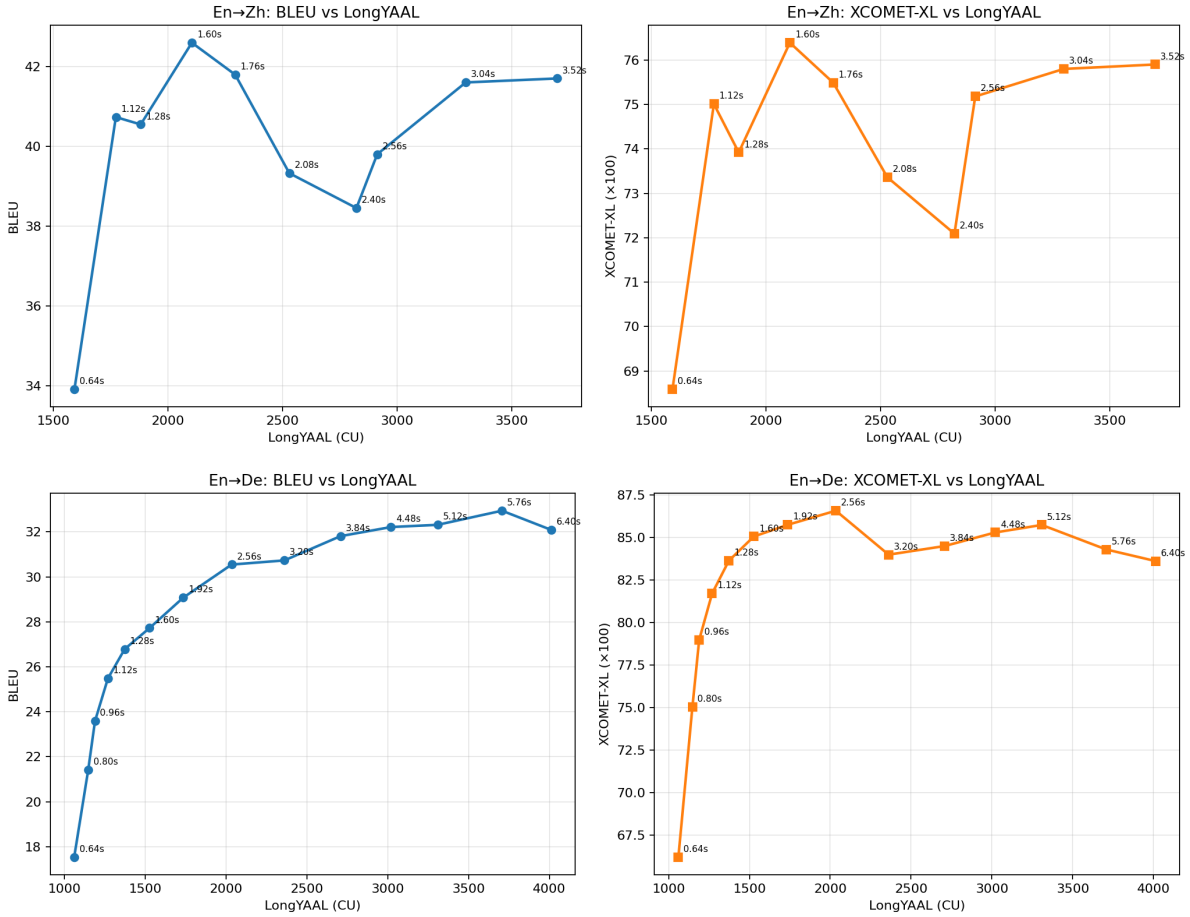

Figure 2: Quality–latency trade-off curves parameterized by chunk_sec. The model gracefully traverses the operating space without retraining.

ever, the sub-track latency on En→De is lower than the main track, indicating that the early commitment of disambiguated entities reduces redundant re-reading.

4.3 Latency Sensitivity and Observations

Sensitivity of chunk_sec. The architectural advantage of internalizing the read/write policy is demonstrated in Figure 2. By sweeping the audio chunk size from 0.64 s to 6.40 s, the model seamlessly traverses the quality–latency trade-off curve. The chosen operating points lie precisely at the natural elbows of these curves, maximizing XCOMET while remaining comfortably below the 2.0 s and 4.0 s limits.

Qualitative Observations. End-to-end testing reveals highly calibrated behavior. Across 919 utterances, empty emissions occur in $\leq 0.4\%$ of cases, strictly limited to ultra-short acoustic tails. Furthermore, adjusting the chunk size does not alter the underlying fraction of <wait> decisions emitted; the model dynamically adjusts its per-step token output instead, proving that the syntax-aware supervision generalizes robustly beyond the static boundaries used during training.

5 Conclusion

We presented the CUHKSZ team submission for IWSLT 2026. By discarding the traditional composite pipeline in favor of a natively aligned multimodal LLM (Qwen3-Omni-30B) and internalizing the read/write policy via syntax-aware supervision, we dramatically simplified streaming inference. Orchestrated by a lightweight Python agent, our system achieves highly competitive computation-aware latency and translation quality on unsegmented speech, while seamlessly extending to zero-shot context injection for the Extra Context subtrack.

6 Acknowledgment

This paper is supported by Project W2531054 of the National Natural Science Foundation of China, and the Program for Guangdong Introducing Innovative and Entrepreneurial Teams.

References

David Ifeoluwa Adelani, Victor Agostinelli, Antonios Anastasopoulos, Luisa Bentivogli, Ondřej Bojar, Sebastien Bratières, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Lizhong Chen, Marcello Federico, Marco Gaido, Mahendra Gupta, HyoJung Han, Ali Hatami, David Javorský, Yejin Jeon, Marek Kasztelnik, Antoine Laurent, and 33 others. 2026. Speech translation and metrics in 2026: Findings of the iwslt campaign. In *Proceedings of the 23rd International Conference on Spoken Language Translation (IWSLT 2026)*, San Diego, California, US. Association for Computational Linguistics.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*, pages 4218–4222.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2023. xCOMET: Transparent machine translation evaluation through fine-grained error detection. *arXiv preprint arXiv:2310.10482*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*.

Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024. LLMs are zero-shot context-aware simultaneous translators. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1207, Miami, Florida, USA. Association for Computational Linguistics.

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.

Sara Papi, Matteo Negri, and Marco Turchi. 2023. Attention as a guide for simultaneous speech translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13340–13356, Toronto, Canada. Association for Computational Linguistics.

Peter Polák, Sara Papi, Luisa Bentivogli, and Ondřej Bojar. 2025. Better late than never: Evaluation of latency metrics for simultaneous speech-to-text translation. *arXiv preprint arXiv:2509.17349*.

Qwen Team. 2025a. Qwen3-Omni technical report. *arXiv preprint arXiv:2509.17765*.

Qwen Team. 2025b. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning (ICML)*, pages 28492–28518.

Seamless Communication, Loïc Barrault, Yu-An Chung, Mariano Cora Meglioli, David Dale, and 1 others. 2023. SeamlessM4T: Massively multilingual and multimodal machine translation. *arXiv preprint arXiv:2308.11596*.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021a. 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 (ACL)*, pages 993–1003.

Changhan Wang, Anne Wu, and Juan Pino. 2021b. CoVoST 2 and massively multilingual speech translation. In *Proc. Interspeech 2021*, pages 2247–2251.

Zeyu Yang, Lai Wei, Roman Koshkin, Xi Chen, and Satoshi Nakamura. 2026. SASST: Leveraging syntax-aware chunking and LLMs for simultaneous speech translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(40):34358–34367.

A Streaming Agent Pseudocode

Algorithm 1 One invocation of the streaming agent.

Require: audio buffer A; end-of-source flag F
Require: state $\mathcal{S} = (turns,\ \ell,\ buf,\ nfail)$
Require: hyper-parameters `chunk_sec`, `min_emit_chars`, `max_turns`, `history_window_sec`, ...

// (a) Read control
if $\neg F$ **then**
  $r \leftarrow$ `chunk_sec` $\cdot\, SR$ **;** **if** $nfail \geq 3$ **then** $r \leftarrow 2r$ ▷ backoff
  if $|A| - \ell < r$ **then return** READ
  end if
end if
$C \leftarrow A[\ell{:}|A|];\ \ell \leftarrow |A|$
// (b) One backbone call
$H \leftarrow$ PRUNEHISTORY$(turns,$ `max_turns`, `history_window_sec`$)$
$M \leftarrow$ BUILDMESSAGES$(\text{systemPrompt}, H, C)$
$y \leftarrow$ vLLM.CHAT(M)
// (c) Intercept the read/write policy
if $y = \varepsilon$ **or** $\neg$ISTARGETLANG(y) **or** $y = turns[-1].trans$ **or** $y \sqsubseteq turns_{\text{recent}}$ **then**
  $nfail \mathrel{+}= 1$; **return** READ
end if
$turns \leftarrow turns \cup \{(C, y)\};\ nfail \leftarrow 0$
$buf \leftarrow buf \oplus y$
// (d) Emission guards
if $|buf| <$ `min_emit_chars` **then return** READ
end if
$(e,\ leftover) \leftarrow$ TRIMTRAILING(buf)
$buf \leftarrow leftover$
if $e = \varepsilon$ **then return** READ
end if
return WRITE$\big($NORMALISE$(e),\ finished = F\big)$

B Data Statistics and Hyperparameters

Table 2: Training corpora statistics and the number of retained chunk-level examples after quality filtering.

Corpus	Hours	Retained Examples
LibriSpeech	960	88 K
Common Voice 17.0 (en)	1,470	74 K
CoVoST2 (en→X)	425	34 K
VoxPopuli (en)	530	24 K
Total (post-filter)	**3,385**	**220 K**

Table 3: LoRA fine-tuning hyperparameters for both En→Zh and En→De models.

Hyperparameter	Value
Target Modules	`q_proj, k_proj, v_proj, o_proj`
LoRA Rank (r)	16
LoRA Alpha (α)	32
LoRA Dropout	0.05
Peak LR (En→Zh)	1×10^{-4}
Peak LR (En→De)	5×10^{-5}
Global Batch Size	128
Training Epochs	3
Hardware Setup	8×NVIDIA A100 GPUs