Pinch-AST: Robust Cascaded Speech Translation System for the IWSLT 2026 Simultaneous Speech Translation task

Carlos Bentes
Pinch
carlos@startpinch.com

Christian Safka
Pinch
christian@startpinch.com

Abstract

We describe Pinch-AST, our submission to the IWSLT 2026 Simultaneous Speech-to-Text Translation shared task, covering all four official directions (En → De, En → It, En → Zh, Cs → En) under both low- and high-latency regimes. Pinch-AST is a cascaded system pairing off-the-shelf speech models with a translation backbone adapted per language pair via LoRA on ASR-noise-augmented parallel data. The streaming policy is a character-level longest-common-prefix retranslation strategy, and the full pipeline runs on a single H100 80 GB GPU within the real-time budget. Evaluated on the IWSLT 2026 development set, Pinch-AST achieves competitive quality–latency trade-offs across all four language pairs in both latency regimes.

1 Introduction

Simultaneous speech translation requires producing target-language text incrementally from a continuous speech stream, balancing translation quality against emission latency. In the IWSLT 2026 edition of the task (Adelani et al., 2026), both constraints are tight: audio is delivered unsegmented and long-form (up to ∼2.5 hours), and systems must run in real time on a single H100 GPU.

This manuscript describes Pinch-AST's submission to the IWSLT 2026 Simultaneous Speech Translation track, covering all four official directions: English → {German, Chinese, Italian} and Czech → English.

Our system is a cascaded pipeline that pairs Qwen3-ASR-1.7B (Shi et al., 2026) and the Qwen3-ForcedAligner-0.6B (Shi et al., 2026) for streaming transcription and utterance-boundary detection with Qwen3.5-4B (Qwen Team, 2026) fine-tuned as a multilingual translator via LoRA adapters covering all four language pairs. Both the ASR and MT stages are governed by Local Agreement (Liu et al., 2020; Polák et al., 2022) on their respective hypotheses, yielding incremental, retraction-free output.

Two data augmentations are central to the system. First, we apply Lexical Noise (Martucci et al., 2021) on the source side of parallel training data, corrupting clean text with an ASR confusion matrix estimated directly from Qwen3-ASR transcriptions of held-out audio. Second, we extend Lexical Noise with prefix-consistent training: word-aligned source/target pairs are truncated at matched prefix boundaries so that the MT model learns to translate incomplete input into incomplete output without speculating beyond the evidence.

Evaluated on the official MCIF (Papi et al., 2026) development set at different segment size configurations, our system improves XCOMET-XL over the IWSLT 2026 organizer baseline by +4.1, +5.7, and +2.6 points on En → De, En → It, and En → Zh respectively, while satisfying the real-time constraint (RTF < 1.0) on a single H100.

2 Task Description

The task requires translating unbounded, unsegmented audio into target-language text incrementally. Test data consists of long-form recordings (up to ∼2.5 hours) of ACL scientific talks for the English-source directions and political conference talks for Czech → English (Adelani et al., 2026). Systems are ranked by translation quality within two latency regimes: low (0–2 s) and high (2–4 s), as measured by non-computation-aware LongYAAL (Polák et al., 2025).

We submit to both regimes in every direction under the constrained-with-LLM data condition. Docker submissions must run on a single NVIDIA H100 GPU with 80 GB HBM, a constraint that directly shapes our architecture. Outputs are produced via the SimulStream (Gaido et al., 2025) toolkit; MCIF is the development set for English-source directions, and the organizers' dedicated

Proceedings of the 23rd International Conference on Spoken Language Translation (IWSLT 2026), pages 268–271

dev-dataset is used for Czech → English.

3 Pinch-AST System Description

3.1 Overview

Pinch-AST is a three-component cascade: *Qwen3-ASR-1.7B* (or *Parakeet-TDT-0.6B-v3* for Cs → En) → *Qwen3.5-4B* + per-pair LoRA (MT, served via `vLLM`) → SimulStream emission layer (character-level LCP, CJK-aware).

Audio arrives as a continuous 16 kHz stream and is processed in fixed-size chunks (640 ms in the low-latency regime, 2500 ms in the high-latency regime). Each chunk triggers at most one ASR pass and one MT pass. Translation emission to the client is incremental and, in the baseline-compatible configuration, append-only (no token deletions, normalized erasure NE = 0). The full pipeline is encapsulated in a `CascadedSpeechProcessor`. It communicates with the SimulStream WebSocket server and is distributed as a single Docker image satisfying the IWSLT 2026 hardware requirements.

3.2 ASR Component

Qwen3-ASR-1.7B (En → X). Qwen3-ASR is a multilingual audio LLM with strong self-correction on partial input, which we found essential for the re-translation streaming paradigm. On each chunk, we feed the model the entire accumulated audio since the last utterance boundary and take its full transcription as the current hypothesis. Because Qwen3-ASR re-decodes the buffer on each call, we apply character-level longest-common-prefix (LCP) across consecutive hypotheses to identify the stable transcript prefix.

Qwen3-ForcedAligner-0.6B (boundary detection, En → X). When the stable transcript contains a sentence-ending punctuation mark, we invoke the forced aligner on the current audio buffer using the stable transcript as reference text. The aligner returns per-word start and end timestamps; we use the timestamp of the boundary punctuation to trim the audio buffer to the point immediately after the sentence ends, thereby freezing the utterance.

Parakeet-TDT-0.6B-v3 streaming (Cs → En). Czech parliamentary and academic speech in our Cs → En development set rarely produces clear sentence-ending punctuation in streaming ASR output. This causes Qwen3-ASR buffers to grow without bound and drives RTF above 5 (i.e., $5x$ slower than real time). Parakeet-TDT-0.6B-v3 is a FastConformer + Token-and-Duration-Transducer model from NVIDIA NeMo that supports cache-aware buffered streaming natively. We configure it with 4-second chunks and 10-second left context.

Parakeet word timestamps provide native boundary information that the Qwen forced aligner cannot supply for Czech. When the streaming decoder fails to emit sentence-ending punctuation, a hybrid rescoring mode overrides Parakeet's boundaries with pause-based heuristics.

3.3 Machine Translation Component

The MT backbone is *Qwen3.5-4B*, a dense 4B-parameter instruction-tuned LLM served through `vLLM` 0.19.1 with PagedAttention, automatic prefix caching, and native LoRA loading. At 4B parameters in `bfloat16`, the backbone occupies approximately 8 GB of GPU memory, leaving sufficient headroom for the KV cache and for co-resident ASR models on the same H100 GPU.

A per-language-pair LoRA adapter ($r = 32$, $\alpha = 64$, ∼24M trainable parameters) is loaded at inference time via `vLLM`'s native LoRA API. The base model is shared across language pairs, while the adapter is swapped at container start, such that each container serves a single translation direction.

3.4 Streaming Policy and Emission

Pinch-AST implements a character-level LCP re-translation policy. On each ASR trigger with newly available stable source text, the MT component re-translates the full stable source sequence from scratch (with prefix caching applied to the system prompt).

The new hypothesis is compared character-by-character against the previous one. Only characters in their longest common prefix (LCP) that extend beyond what has already been emitted are released.

4 Data Augmentation

4.1 Noise model

We apply the Lexical Noise model of Martucci et al. (Martucci et al., 2021), which models ASR as a per-word IDS (Insertion–Deletion–Substitution) channel whose conditional distributions $P(a_j \mid c_i)$ are estimated directly from a transcription–reference alignment on a held-out audio sample. Concretely: 1) Transcribe a held-out sample of English training audio with `Qwen3-ASR-1.7B` and a held-out sample of Czech audio with

Parakeet-TDT-0.6B-v3; 2) Word-align ASR output to reference transcripts to build a per-source-language confusion matrix: for each clean word, estimate the distribution over its ASR-time substitutions, along with a deletion rate and an insertion distribution $P(a_j \mid \phi)$; 3) For each source-side sentence in the clean MT training data, apply the IDS channel one or more times with different random seeds to synthesize corrupted variants; 4) Build the fine-tuning mixture as clean originals plus corrupted copies, preserving the target side unchanged.

We assume the Lexical Noise model is domain-independent: noise patterns estimated on one audio domain transfer to another without re-estimation, so one confusion matrix per source language (English, Czech) suffices for all four pairs.

4.2 Prefix-consistent training data

In addition to full-sentence corrupted pairs, we construct prefix-truncated training examples to explicitly teach the model to translate from partial input: 1) Word-align each clean (source, target) pair with awesome-align (Dou and Neubig, 2021); 2) Build a monotonic alignment envelope: for each source-word prefix, find the largest target-word prefix such that no cross-alignment crosses the cut; 3) Truncate each parallel pair at 30%, 50%, 70% of its source length under this envelope, plus the full pair (100%); 4) Apply Martucci IDS noise augmentation to the truncated pairs; 5) Mix full-length and prefix-truncated examples in an approximately $50/50$ ratio per language pair.

5 Training

All fine-tuning runs are conducted on a single node with $8\times$ NVIDIA H100 80 GB HBM3, using HuggingFace accelerate with a DDP launch configuration per-pair. At inference time, the constrained track budget is a single H100 80 GB, so all training-time decisions that affect deployable memory footprint are made with the single-H100 budget in mind. The submitted system uses per-language-pair LoRA adapters.

6 Experiments

All evaluation runs are executed inside the same Docker image used for submission, with SimulStream 0.3.0 driving the client side and OmniSTEval 0.1.7 performing scoring. Translation quality is reported using OE-COMET (Unbabel/wmt22-comet-da, the organizer-specified COMET model), chrF, and OE-BLEU. Latency is reported as LongYAAL-CU (non-computation-aware, the primary ranking metric used by the organizers) and LongYAAL-CA (computation-aware). Reference datasets include MCIF long-form (21 talks) for En→X and the IWSLT 2026 CS development set (43 recordings) for Cs→En.

Our experiments focused on balancing translation quality with aggressive latency targets across all language pairs. We observed that while initial settings of 960 ms segment size chunks provided good COMET scores for English-to-X translations, they resulted in prohibitive latencies for English-to-Chinese and Czech-to-English, often exceeding 4 seconds. By shifting to 640 ms chunk size and eliminating history buffers, we successfully brought latencies under the 2-second threshold.

With the per-pair LoRA system selected, we ran the final submission sweep at four chunk sizes—640 ms, 960 ms, 1600 ms, and 2500 ms—across all four language pairs. The 640 ms and 960 ms configurations fall within the low-latency regime (target LongYAAL-CU 0–2 s), while 1600 ms and 2500 ms fall within the high-latency regime (target 2–4 s). We submitted all four configurations per pair and used development-set LongYAAL to determine the regime assignment.

7 Results

Table 1 lists the complete dev-set results submitted for both latency regimes across all four language pairs. Latency is reported as LongYAAL-CU (primary ranking metric) and LongYAAL-CA (computation-aware). Quality is reported as OE-COMET (Unbabel/wmt22-comet-da), chrF, and OE-BLEU. Datasets are MCIF long-form (En→X, 21 talks) and the IWSLT 2026 CS development set (43 recordings).

8 Conclusion

We have described Pinch-AST, a cascaded simultaneous speech translation system for IWSLT 2026 built from three off-the-shelf speech models (Qwen3-ASR-1.7B, Qwen3-ForcedAligner-0.6B, Parakeet-TDT-0.6B-v3) and one lightly adapted LLM (Qwen3.5-4B with per-pair LoRA), wired together with a character-level LCP re-translation policy. The system covers all four official directions in both latency regimes, fits within a single H100

Chunk	Regime	LY-CU	COMET	chrF	BLEU
En→De					
640	low	1639	0.7380	59.8	25.6
960	low	2153	0.7604	61.2	28.6
1600	high	2694	0.7564	61.5	29.1
2500	high	3035	0.7655	61.8	29.4
En→It					
640	low	1594	0.7820	67.1	38.3
960	low	2094	0.7968	68.5	41.4
1600	high	2660	0.7992	68.8	41.9
2500	high	3103	0.8052	68.6	41.6
En→Zh					
640	low	1397	0.8083	32.8	36.0
960	low	1906	0.8122	35.1	38.1
1600	high	2539	0.8132	36.2	39.5
2500	high	2914	0.8147	36.0	38.9
Cs→En					
640	low	1625	0.6515	49.4	17.8
960	low	1475	0.6538	49.3	17.9
1600	high	2350	0.6832	51.7	22.0
2500	high	2468	0.7170	53.6	24.1

Table 1: Pinch-AST dev-set results across all language pairs and chunk sizes. LY is reported in ms.

`80 GB` at inference time, and achieves OE-COMET scores of 0.652–0.812 in the low-latency regime and 0.717–0.815 in the high-latency regime.

Limitations. The current system makes no use of the context sub-track's ACL paper PDFs, and we did not submit to the context sub-track (nor to the Cs→En Linguistic Mondays sub-track). Our Cs→En pipeline's reliance on `Parakeet` streaming combined with hybrid pause-rescoring is a workaround for the absence of a Czech-capable forced aligner; a streaming-native Czech ASR with reliable sentence boundary detection would likely reduce latency while improving translation quality.

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.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128. Association for Computational Linguistics.

Marco Gaido, Sara Papi, Mauro Cettolo, Matteo Negri, and Luisa Bentivogli. 2025. Simulstream: Open-source toolkit for evaluation and demonstration of streaming speech-to-text translation systems. *arXiv preprint arXiv:2512.17648*.

Danni Liu, Gerasimos Spanakis, and Jan Niehues. 2020. Low-latency sequence-to-sequence speech recognition and translation by partial hypothesis selection. In *Proc. Interspeech 2020*, pages 3620–3624.

Giuseppe Martucci, Mauro Cettolo, Matteo Negri, and Marco Turchi. 2021. Lexical modeling of ASR errors for robust speech translation. In *Proc. Interspeech 2021*, pages 2282–2286, Brno, Czechia.

Sara Papi, Maike Züfle, Marco Gaido, Beatrice Savoldi, Danni Liu, Ioannis Douros, Luisa Bentivogli, and Jan Niehues. 2026. MCIF: Multimodal crosslingual instruction-following benchmark from scientific talks. In *The Fourteenth International Conference on Learning Representations (ICLR)*.

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*.

Peter Polák, Ngoc-Quan Pham, Tuan Nam Nguyen, Danni Liu, Carlos Mullov, Jan Niehues, Ondřej Bojar, and Alexander Waibel. 2022. CUNI-KIT system for simultaneous speech translation task at IWSLT 2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 277–285, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents. `https://qwen.ai/blog?id=qwen3.5`.

Xian Shi, Xiong Wang, Zhifang Guo, Yongqi Wang, Pei Zhang, Xinyu Zhang, Zishan Guo, Hongkun Hao, Yu Xi, Baosong Yang, Jin Xu, Jingren Zhou, and Junyang Lin. 2026. Qwen3-ASR technical report. *arXiv preprint arXiv:2601.21337*.