RASST: Retrieval-Augmented Simultaneous Speech Translation

Anonymous ACL submission

Abstract

Simultaneous speech translation produces target text incrementally from partial speech input. Recent speech large language models have markedly improved SST quality but still struggle with rare and domain-specific terminology. Retrieval augmentation has helped in automatic speech recognition and neural machine translation, but extending it to SST is non-trivial: retrieval must be fast and accurate under partial speech, and the model must decide whether and when to apply retrieved terms during incremental generation. We propose **R**etrieval-**A**ugmented **S**imultaneous **S**peech **T**ranslation (RASST), which addresses both challenges. For accurate cross-modal retrieval under partial input, RASST trains a lightweight speech–text retriever that produces chunkwise terminology hints for the Speech LLM via multi-scale retrieval. To use these hints correctly, we synthesize training data that teaches the Speech LLM to decide whether and when to apply each retrieved term. Experiments on ACL 60/60 dev set and the ESO test set show that RASST improves terminology accuracy by nearly 40% and overall translation quality by up to 3 BLEU points, with negligible computational overhead.

1 Introduction

Simultaneous Speech Translation (SST) aims to generate translations incrementally as partial speech input is received. It has wide applications in international conferences and cross-lingual conversations (Ma et al., 2020b; Ren et al., 2020). Recently, Large Language Models (LLMs) have shown strong potential as backbones for SST systems (Wang et al., 2025; Ouyang et al., 2025b; Fu et al., 2025; Cheng et al., 2025). However, they still struggle to accurately translate domain-specific terminologies, including technical jargon, person and location names, and abbreviations. As shown in Table 1, a plain SST system translates "Shinzo Abe" incorrectly as "新三·阿贝", which is a phonetic transliteration rather than the official translation of the politician's name.

Source:	Shinzo Abe served as the Prime Minister of Japan.
Reference:	安倍晋三曾担任日本首相。
Plain SST:	新三·阿贝曾担任日本首相。
RASST:	安倍晋三曾担任日本首相。

Table 1: A case showing plain SST without retrieval fails to translate terminology correctly while SST with retrieval is able to do so. English (blue), correct Chinese terms (green), and incorrect ones (red).

Human simultaneous interpreters often rely on external glossaries to translate specialized terminology in real time (Jiang, 2013), and retrieval augmentation has been explored widely in automatic speech recognition (ASR) and neural machine translation (NMT) to improve terminology handling as the glossary size grows (Khandelwal et al., 2021; Huber et al., 2021; Jayanthi et al., 2023; Li et al., 2024, 2025; Wu et al., 2025; Gong et al., 2025; Kong et al., 2025). However, retrieval-augmented SST poses unique challenges. Beyond requiring a cross-modal retriever that is fast and accurate under partial speech input, SST translations, unlike ASR transcripts which are time-aligned with the speech, lag behind the speech. The model must therefore decide not only whether to use a retrieved term, but also when to apply it during incremental translation.

To address these challenges, we propose **R**etrieval-**A**ugmented **S**imultaneous **S**peech **T**ranslation (RASST). RASST trains a lightweight speech–text retriever on variable-duration speech contexts with a contrastive loss, and performs multi-scale inference for accurate retrieval under partial input. We further synthesize training data that teaches the Speech LLM to select relevant retrieved terms and decide when to introduce them during generation, conditioned on the

prior speech context and previously generated translation. Experiments on the ACL 60/60 dev set and ESO test set show that RASST improves terminology translation accuracy by nearly 40% percent and BLEU by up to 3 points on three language directions (En→Zh/De/Ja), while adding negligible computational overhead.

2 Related Works

SST with LLM Recent work shows that simultaneous speech translation (SST) achieves markedly higher quality when large language models (LLMs) serve as the backbone (Ahmad et al., 2024). Koshkin et al. (2024) adapt an LLM to SST via finetuning on synthetic translation trajectories, and Ouyang et al. (2025b) scale this idea with an interleaving architecture for unbounded speech streams. Fu et al. (2025) explore alternative training strategies and data-synthesis pipelines, while Guo et al. (2025) unify streaming transcription and translation with a truncation mechanism that prunes history based on transcribed inputs. However, these systems still struggle with domain-specific terminology and RASST addresses this by integrating a terminology retriever directly into the SST process.

Retrieval-Augmented NMT Retrieval is an effective remedy for translating rare words, named entities, and domain-specific terminology in NMT. Early approaches retrieve translation pieces from similar training pairs (Zhang et al., 2018), mine bilingual terminology with bilingual embeddings (Huck et al., 2019), or fuse dictionary entries into NMT via a Pointer-Disambiguator-Copier (Zhang et al., 2021). kNN-MT (Khandelwal et al., 2021) augments NMT with nearest-neighbor lookup over a large datastore, and follow-up work improves its robustness (Jiang et al., 2021) and efficiency (Meng et al., 2022; Zhu et al., 2023). Other approaches extract and attend over entity candidates (Zeng et al., 2023) or combine retrieval and demonstration with LLMs (Li et al., 2025). For speech translation, Li et al. (2024) prepend retrieved speech segments as demonstrations, and Wu et al. (2025) use sliding-window retrieval to inject terms into a speech-LLM prompt. All of these target offline translation, whereas RASST integrates retrieval directly into incremental streaming translation.

Contextual-Biasing ASR Recognizing rare words and domain-specific terminology, commonly known as contextual biasing, is also well studied in ASR. Early work splits into two lines: joint optimization of the ASR model with contextual n-gram embeddings (Pundak et al., 2018; Jain et al., 2020), and shallow fusion of biased words into end-to-end models (Zhao et al., 2019), later extended with prefix-trie-based deep biasing (Le et al., 2021a) and unified via deep shallow fusion and trie-constrained biasing (Le et al., 2021b). Huber et al. (2021) support runtime addition of context words through an external memory accessed at decoding. To scale biasing lists from hundreds to tens of thousands of entries, subsequent work leverages neural associative memories (Munkhdalai et al., 2022, 2023; Wu et al., 2023, 2024) and retrieval augmentation (Li et al., 2021; Yusuf et al., 2023; Jayanthi et al., 2023). More recent efforts build on speech foundation models. Yang et al. (2024) use coarse CTC decoding to filter hotwords into an LLM-based ASR prompt, and Gong et al. (2024) inject bias words into a SpeechLLM via prompt formats and a biasing-fusion network. BR-ASR (Gong et al., 2025) decouples bias retrieval from ASR decoding through contrastive learning and curriculum training, scaling to 200k entries at 20ms retrieval latency, while GLCLAP (Kong et al., 2025) retrieves bias words directly from speech via global-local contrastive language-audio pre-training, enabling plug-in biasing on frozen models like Whisper. Finally, PAC (Fu et al., 2026) combines pronunciation-guided context learning with pronunciation-discriminative reinforcement learning. Unlike ASR, where the transcript is time-aligned with the speech, SST translations lag behind the input, so the model must therefore decide whether and when to apply each retrieved term.

3 Problem Formulation

Let $\boldsymbol{s} = (s_1, s_2, \ldots, s_T) \in \mathbb{R}^T$ denote the source speech waveform, and let the glossary $\mathcal{G} = \{(e_j, e_j^{\text{trans}})\}_{k=1}^{|\mathcal{G}|}$, where e_j is a term in the source language and e_j^{trans} is its translation. At each step i, a fixed-duration speech chunk is received

$$\boldsymbol{s}_i = \big(s_{i \cdot l+1}, \ldots, s_{(i+1) \cdot l}\big), \tag{1}$$

where l is the chunk length.

Given the accumulated speech chunks $\boldsymbol{s}_{1:i}$, the retriever R_ϕ selects a subset of relevant terminol-

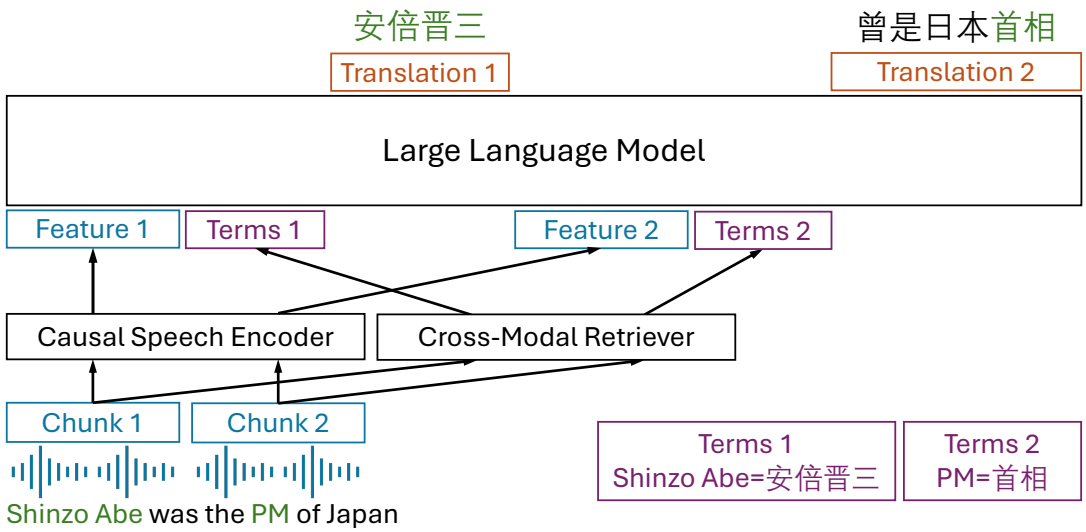

Figure 1: Overview of RASST. RASST interleaves speech chunks, retrieved terms, and translation outputs. For each incoming speech chunk, a causal speech encoder produces its feature representation, and a cross-modal retriever returns the corresponding terms. The retrieved terms are appended to the speech features and passed to the large language model, which decodes the next translation segment.

ogy entries

$$G_i = R_\phi(\boldsymbol{s}_{1:i}, \mathcal{G}) \subseteq \mathcal{G}. \quad (2)$$

Conditioned on the observed speech $\boldsymbol{s}_{1:i}$, the previously generated translation tokens $\boldsymbol{y}_{1:i-1}$, and the retrieved terminology context $G_{1:i}$, the model $\pi_\theta(\boldsymbol{s}_{1:i}, \boldsymbol{y}_{1:i-1}, G_{1:i})$ produces a partial translation $\boldsymbol{y}_i$ at step i. The output $\boldsymbol{y}_i = (y_1^i, \ldots, y_{|\boldsymbol{y}_i|}^i)$ may contain a variable number of tokens.

If $|\boldsymbol{y}_i| = 0$, the model chooses to wait for additional speech input without emitting any translation. Otherwise, when $|\boldsymbol{y}_i| > 0$, the model outputs a partial translation. All tokens in $\boldsymbol{y}_i$ are assigned an identical delay of $i \cdot l$, corresponding to the elapsed time when the speech chunk $\boldsymbol{s}_i$ becomes available.

Finally, translation quality and latency are evaluated with respect to the full source waveform $\boldsymbol{s}$, the aggregated translation hypothesis $\boldsymbol{y}$, the reference translation $\boldsymbol{y}$, the source transcript $\boldsymbol{x}$, and the delay associated with each generated token.

4 Retrieval-Augmented Simultaneous Speech Translation

In this section, we first present an overview of RASST. We then describe the cross-modal retriever design and the training procedure that enables the Speech LLM to effectively leverage retrieved terms during simultaneous translation.

4.1 RASST Overview

Figure 1 presents an overview of RASST. We adopt the interleaving SST formulation of InfiniSST (Ouyang et al., 2025b), which alternates between incoming speech chunks and incremental translation outputs, and RASST augments this with a retrieval step:

$$\boldsymbol{y}_i \sim \text{SpeechLLM}(\boldsymbol{s}_1, G_1, \boldsymbol{y}_1, \ldots, \boldsymbol{s}_i, G_i). \quad (3)$$

After each chunk $\boldsymbol{s}_i$ arrives, the retriever returns candidate terms G_i likely to appear in it, and we append these terms along with their target-language translations to the chunk's representation. Conditioned on the speech context and the retrieved terms, the LLM then generates the next partial translation $\boldsymbol{y}_i$.

4.2 Cross-Modal Retriever

Architecture We adopt a dual-encoder architecture. For text, we use the pre-trained BGE-M3 encoder (Chen et al., 2024) to map each source-language term e_j to a feature vector $\boldsymbol{g}_j \in \mathbb{R}^d$. For speech, we use the Audio Transformer of Qwen3-Omni (Yang et al., 2025) followed by a linear projection to the same dimension d. Since terms are typically short, the speech encoder reads only the current chunk $\boldsymbol{s}_i$ together with a look-back window of duration t_{lb}, rather than the full history $\boldsymbol{s}_{1:i}$ as in Equation 2. We denote this input as $\boldsymbol{s}'_i$ and the resulting features as $\boldsymbol{f}_i \in \mathbb{R}^{m \times d}$, with m as the number of frames in $\boldsymbol{s}'_i$.

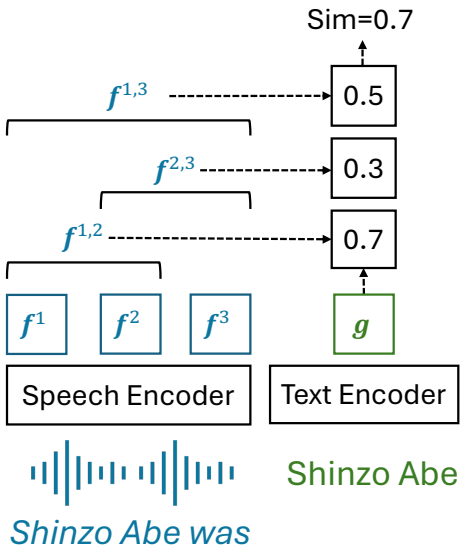

Figure 2: An example of multi-scale inference. The text term is encoded into a single feature vector $\boldsymbol{g}$, while the speech context is encoded into frame features $\boldsymbol{f}^1, \boldsymbol{f}^2, \boldsymbol{f}^3$. We enumerate all windows $(1,2), (2,3), (1,3)$ and compute the corresponding average-pooled features $\boldsymbol{f}^{1,2}, \boldsymbol{f}^{2,3}, \boldsymbol{f}^{1,3}$. The likelihood that the term appears in the speech context is then the maximum cosine similarity between $\boldsymbol{g}$ and any window feature.

Multi-Scale Inference Because terms vary in duration, representing the speech context with a single average-pooled vector loses information. Instead, we pool over windows at multiple scales. As shown in Figure 2, given $\boldsymbol{f}_i = (\boldsymbol{f}_i^1, \boldsymbol{f}_i^2, \ldots, \boldsymbol{f}_i^m)$, for each window $1 \le u < v \le m$ we compute

$$\boldsymbol{f}_i^{u,v} = \text{AvgPool}(\boldsymbol{f}_i^u, \ldots, \boldsymbol{f}_i^v) \in \mathbb{R}^d. \quad (4)$$

The similarity between the speech $\boldsymbol{s}_i'$ and a term e_j is then the maximum cosine similarity over all windows:

$$\text{Sim}(\boldsymbol{s}_i', e_j) = \max_{1 \le u < v \le m} \text{CosSim}(\boldsymbol{f}_i^{u,v}, \boldsymbol{g}_j), \quad (5)$$

and the retrieved terms are

$$G_i = \text{Top-}K_j \left(\text{Sim}(\boldsymbol{s}_i', e_j)\right) \subseteq \mathcal{G} \quad (6)$$

In practice, we build a FAISS index (Douze et al., 2024) over the term vectors $\{\boldsymbol{g}_j\}_{j=1}^{|\mathcal{G}|}$, then issue a Top-$K$ query for each window feature $\boldsymbol{f}_i^{u,v}$, producing a candidate pool of size up to a multiple of K. We perform a second Top-K selection over this pool by similarity score, keeping the highest score per term when duplicates occur, and finally discard any candidate whose similarity falls below a threshold τ to suppress noise.

Data Synthesis Retriever training requires paired speech and terminology data $(\boldsymbol{s}', e_{\text{pos}})$. Since terminology-rich speech corpora are scarce, we synthesize training data from two sources: existing ASR datasets and a terminology database.

For ASR data, we obtain word-level timestamps for each transcript using the Montreal Forced Aligner (MFA)[1], then extract noun phrases as candidate terms using spaCy[2] and normalize their surface forms. We use noun phrases because most domain-specific terminology takes this form and they are abundant in standard transcripts. For each term, we carve out a speech context $\boldsymbol{s}'$ of variable duration surrounding it, which gives the retriever flexibility across context lengths at inference time.

For each term in the terminology database, we prompt Gemini-2.5-Flash[3] to generate several sentences containing the term and synthesize the speech with CosyVoice 3 (Du et al., 2025). The resulting speech is then processed by the same MFA pipeline as the ASR data.

Training Given paired data $(\boldsymbol{s}', e_{\text{pos}})$, we train the retriever with the InfoNCE loss (van den Oord et al., 2019). In addition to in-batch negatives, we mine hard negatives by using the retriever checkpoint, which is updated every 50 training steps, to retrieve the Top-N most similar terms from the entire training set for each positive e_{pos}. Let E_{neg} be the negative set and $E_{\text{all}} = \{e_{\text{pos}}\} \cup E_{\text{neg}}$. The retriever loss $\mathcal{L}_{\text{retriever}}$ is then

$$-\log \frac{\exp(\text{Sim}(\boldsymbol{s}_i', e_{\text{pos}})/\gamma)}{\sum_{e \in E_{\text{all}}} \exp(\text{Sim}(\boldsymbol{s}_i', e)/\gamma)}, \quad (7)$$

where γ is the temperature. Since the timestamp of e_{pos} within $\boldsymbol{s}'$ is known during training through MFA, we bypass the maximum over windows in $\text{Sim}(\cdot)$ and instead set (u, v) to be the smallest window enclosing e_{pos}.

4.3 Speech LLM Training

To teach the Speech LLM to effectively use the retrieved terms G_i during simultaneous translation, we synthesize a corresponding training corpus. Given the speech chunks $\boldsymbol{s}_{1:i}$, we follow InfiniSST's procedure to produce the translation segments $\boldsymbol{y}_{1:i}$. The remaining piece beyond InfiniSST

[1] https://github.com/MontrealCorpusTools/Montreal-Forced-Aligner

[2] https://spacy.io/ and en_core_web_trf model

[3] https://ai.google.dev/gemini-api/docs/models/gemini-2.5-flash

is the retrieved terms $G_{1:i}$. A naive approach is to reuse the noun-phrase terms from retriever training and always populate G_i with the ground-truth term. This, however, would leave the model brittle to retriever mistakes at inference time. To reduce this training–inference mismatch, we build a glossary over all noun phrases in the training set and run the cross-modal retriever from Section 4.2 on every speech chunk. Then we add the retrieved distractors in G_i alongside the ground-truth term. The Speech LLM thus learns to translate in the presence of realistic retriever noise.

A training example is shown in Figure 13. We wrap the ground-truth term's translation in a `<term>. . . </term>` tag, encouraging the Speech LLM to copy ground-truth term translations directly from G_i. We fine-tune with a standard cross-entropy objective applied only to translation tokens $\boldsymbol{y}$. Because translations naturally lag behind speech in the training data, the model also receives implicit supervision for *timing*: it observes when a term appears in the speech context and when its translation appears later in the target sequence.

5 Experimental Setups

5.1 Data

Retriever Training We synthesize the retriever training data from GigaSpeech (Chen et al., 2021) and Wikidata[4]. GigaSpeech is a multi-domain English ASR corpus containing 10K hours of high-quality labeled audio. We process it following the procedure in Section 4.2, yielding 4.7M training pairs. The variable speech-context duration is sampled uniformly from $2.88, 3.84, 4.80, 5.76$ seconds, corresponding to the inference-time chunk sizes $0.96, 1.92, 2.88, 3.84$ plus a $t_{\text{lb}} = 1.92$-second look-back. The look-back captures terms that cross chunk boundaries and we set it to 1.92 seconds since most terms are shorter than this, as shown in Figure 6.

For Wikidata, we extract candidate terms from the RDF truthy dump, keeping entities that have both an English `rdfs:label` and at least one `wdt:P31` triple. Full processing details are given in Appendix A. For each retained term, we prompt Gemini-2.5-Flash to generate three natural utterances of 5–15 words containing the term (Figure 14). We then synthesize the utterances with CosyVoice 3[5], using speaker prompts randomly sampled from GigaSpeech. This results in 2.9M training pairs.

Speehc LLM Training We synthesize the Speech LLM training data using the same GigaSpeech ASR dataset. The interleaved En-Zh/De/Ja translation data is synthesized similar to InfiniSST with details in Appendix B. We subsample 12.5K instances for Speech LLM fine-tuning. We build a 140K glossary from GigaSpeech and Wikidata for distractor retrieval, and cap the size of G_i at 20 terms.

Evaluation We evaluate on two datasets. The first is the ACL 60/60 dev set (Salesky et al., 2023), which contains five English ACL talks translated into 10 target languages. We evaluate on the En-Zh/De/Ja directions, feeding the unsegmented talks directly to the model. This dataset includes human-annotated terminology, which we use to construct the glossary for the model to retrieve from. After deduplication, the glossary contains 238 unique terms. The second is the ESO test set (Iranzo-Sánchez et al., 2025), which consists of five long-form pre-recorded English oncology training videos provided by the European School of Oncology, with translations into French, Spanish, German, and Slovene. We use GPT-5.4[6] to supplement the dataset with Chinese and Japanese translations, and to extract a candidate terminology list. After manual verification, the resulting glossary contains 217 terms. To test robustness under larger glossaries, we expand both glossaries to 1K and 10K terms using Wikidata terms from domains similar to ACL or oncology and ensure that all added terms are unseen during training.

5.2 Evaluation Metrics

Latency Following IWSLT 2025 practice (Abdulmumin et al., 2025), we use StreamLAAL (Papi et al., 2024) as our latency metric. StreamLAAL segments the model hypothesis to align with reference sentences and reports the average latency over the resulting (hypothesis, reference) pairs.

Translation Quality We evaluate translation quality on the aligned (hypothesis, reference) sentence pairs using SacreBLEU (Papineni et al., 2002; Post, 2018).

Terminology Accuracy We introduce terminology accuracy to measure how accurately the model

[4] https://www.wikidata.org/

[5] FunAudioLLM/Fun-CosyVoice3-0.5B-2512

[6] https://developers.openai.com/api/docs/models/gpt-5.4

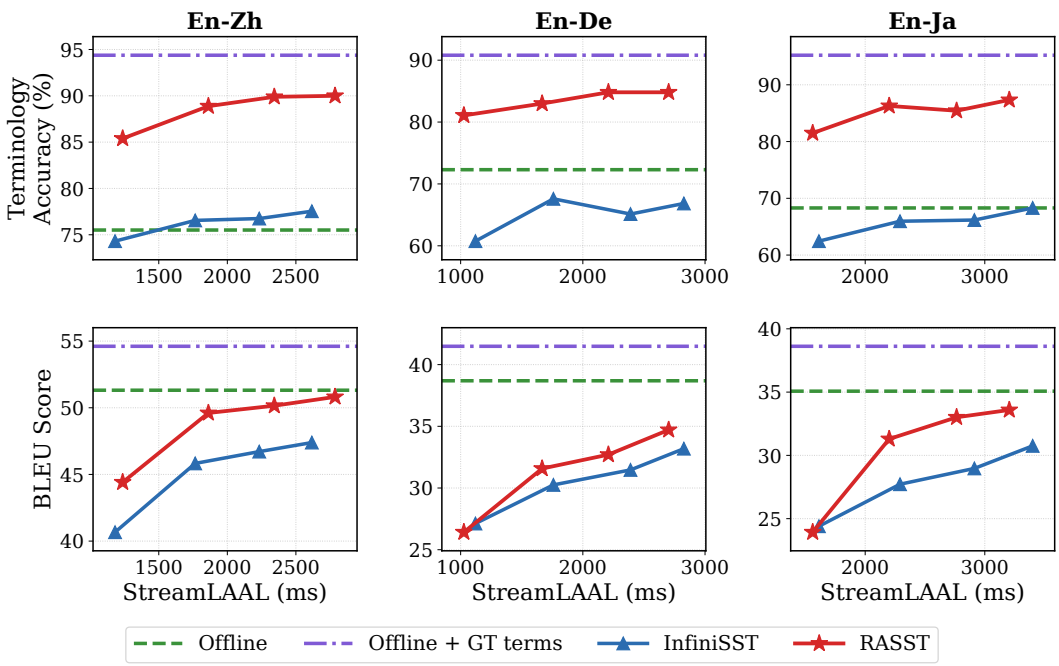

Figure 3: Quality–latency trade-off of RASST and baselines on the ACL 60/60 dev set. Quality is measured by both terminology accuracy and BLEU. Across all three language directions, RASST improves terminology accuracy over InfiniSST by at least 10% while also achieving higher translation quality.

translates terms. For each terminology occurrence in the reference, we check whether its translation appears as an exact match in the aligned hypothesis. The metric is the percentage of such occurrences correctly matched, ranging from 0 to 100%.

5.3 Model Configuration

Architecture For the retriever, we use a Qwen3-Omni Audio Transformer[7] as the speech encoder, followed by a linear projection. For text encoding, we use the dense BGE-M3[8] encoder and extract the feature of [CLS] token. For the Speech LLM, we use Qwen3-Omni-30B-A3B-Instruct[9].

Training We train both the cross-modal retriever and the Speech LLM with LoRA (Hu et al., 2022). For retriever training, the number of hard negative N is 1024 and the global batch size is 8192. For Speech LLM training, we use sequence packing and a global batch size of 8192 tokens. Full hyper-parameters and details are provided in Appendix D.

Inference We implement the inference agent in SimulEval (Ma et al., 2020a), varying the chunk size l over $0.96, 1.92, 2.88, 3.84$ seconds. The Speech LLM is invoked once per chunk and generates up to 40 new tokens per call. We serve it with vLLM (Kwon et al., 2023) using tensor parallelism of size 2, decode with temperature 0.6, top-p=0.95, and top-k=20, and strip the assistant's `<term>` tags from the hypothesis before scoring.

For retrieval, we build a FAISS inner-product index over ℓ_2-normalized term embeddings. At each step, the retriever encodes the current speech chunk together with a 1.92-second look-back, and queries the index using only those windows whose right boundary lies within the current chunk. We use Top-K=10 and discard candidates with similarity below τ=0.78[10].

5.4 Baselines

Offline We use Qwen3-Omni-30B-A3B-Instruct to translate pre-segmented speech utterances offline, with the same decoding parameters as RASST. This serves as an upper bound on translation quality.

Offline + GT Terms We extend the **Offline** baseline by adding the ground-truth terms appearing in each utterance to the prompt in a manner similar to Gong et al. (2024), helping the model handle terminology correctly. This serves as an upper bound

[7] https://huggingface.co/Atotti/Qwen3-Omni-AudioTransformer

[8] https://huggingface.co/BAAI/bge-m3

[9] https://huggingface.co/Qwen/Qwen3-Omni-30B-A3B-Instruct

[10] We select τ on the dev set to allow at most a 1% drop in recall.

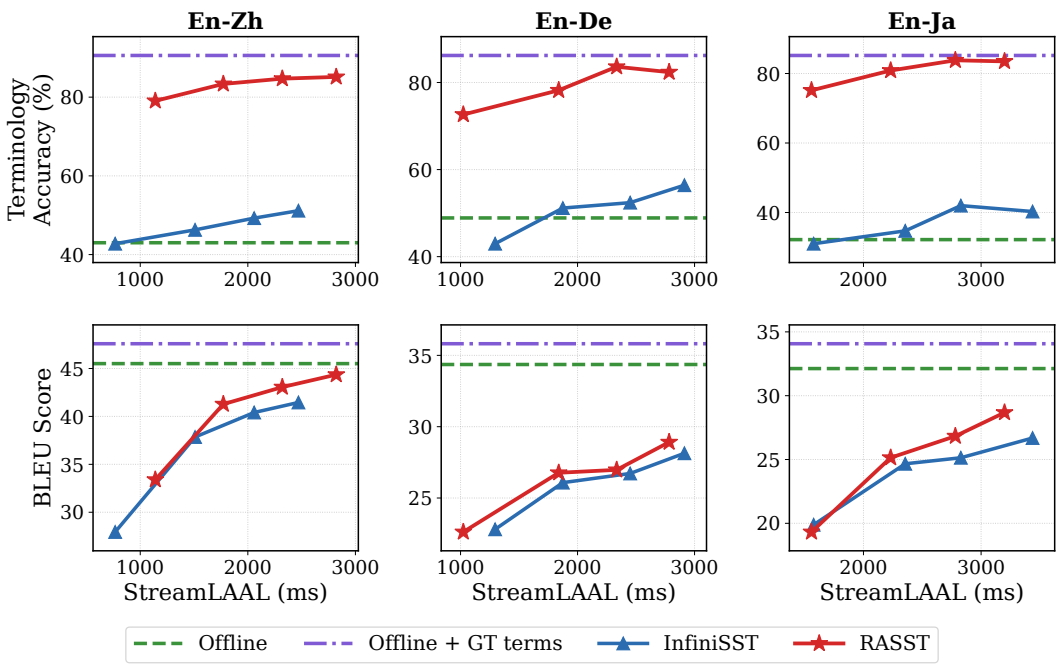

Figure 4: Quality–latency trade-off of RASST and baselines on the ESO test set. Quality is measured by both terminology accuracy and BLEU. Across all three language directions, RASST improves terminology accuracy over InfiniSST by nearly 40% while also achieving slightly higher translation quality.

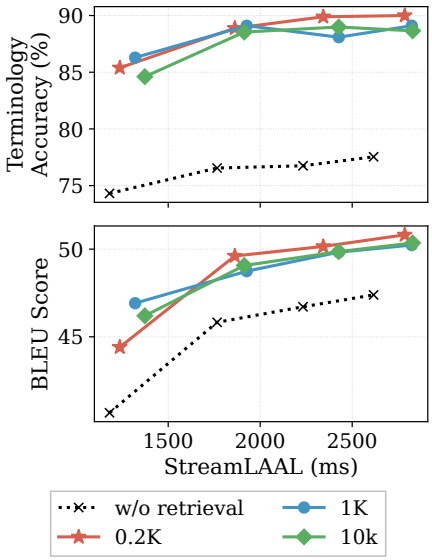

Figure 5: Scaling the glossary from 0.2K to 1K and 10K terms. RASST shows only a slight drop in terminology accuracy and translation quality, remaining well above the no-retrieval baseline.

on terminology translation accuracy.

InfiniSST InfiniSST (Ouyang et al., 2025b) is a strong SST baseline for simultaneous translation of unbounded speech and achieved top performance in the low-latency track at IWSLT 2025 (Ouyang et al., 2025a; Abdulmumin et al., 2025). We train InfiniSST on the same data and with the same Speech LLM as RASST, but remove all terminology input G_i. This setting can be viewed as RASST without retrieval.

6 Results and Analysis

6.1 Main Results

RASST substantially improves terminology accuracy As shown in the first row of Figures 3 and 4, RASST consistently improves terminology accuracy over the strong InfiniSST baseline, by at least 10% on the ACL 60/60 dev set and nearly 40% on the ESO test set. ESO sees a larger gain because its oncological terms are hard to recognize and translate, whereas many of the annotated terms in ACL 60/60 are everyday words like "words", "model", and "weights" which are easy to translate correctly. Notably, RASST even surpasses the *Offline* baseline, indicating that the problem cannot be solved simply by providing full speech context. A gap to *Offline + GT Terms* remains because the retriever is imperfect, but it is typically under 5%, underscoring RASST's effectiveness.

RASST improves overall translation quality As shown in the second row of Figures 3 and 4, RASST also improves overall translation quality over InfiniSST. The BLEU gains are smaller than the gains in terminology accuracy, which is expected since terms occur relatively infrequently in

Chunk Size (s)	0.96	1.92	2.88	3.84
Wall-Clock Time (s)	0.036	0.042	0.042	0.043

Table 2: Retriever computation cost across speech chunk sizes. The retriever's overhead is negligible.

the speech.

RASST is robust to large glossary size Figure 5 shows that RASST remains competitive as the glossary scales to 10K terms. Going from 0.2K to 1K and then 10K terms gradually reduces terminology accuracy by less than 2% and BLEU by less than 1 point, and RASST still clearly outperforms the no-retrieval baseline.

RASST is computationally efficient Table 2 reports the additional computational cost introduced by the retriever. We run evaluation on two NVIDIA RTX A6000 GPUs, serving the Speech LLM with vLLM at tensor parallelism 2 and `gpu_memory_utilization`=0.72, while the retriever encoder fits in the remaining GPU memory. The retriever's wall-clock time per chunk is at most 3.75% of the chunk size, and becomes smaller as the chunk grows, which makes it negligible in practice.

6.2 Ablation Studies

Retriever hard negatives and threshold We examine the effect of training-time hard negatives and the inference-time threshold τ. In addition to the default retriever trained with N=1024 hard negatives, we train two variants with N=256 and N=0. We evaluate their precision and recall on the GigaSpeech dev set against three glossaries: a 1K glossary built from GigaSpeech noun phrases, and 10K and 100K extensions with Wikidata terms unseen during training. For each (retriever variant, glossary) pair, we sweep τ from 0 to 1 to obtain a precision-recall curve.

Figure 10 shows the results. First, increasing N yields clearly higher recall at the same precision, confirming the value of hard-negative mining. Second, recall degrades as the glossary grows. Based on these curves, we set τ=0.78 at inference: it costs at most a 1% drop in recall while roughly reducing the retrieval noise by half. Compared with $\tau = 0$ in Figure 11, $\tau = 0.78$ achieves higher terminology accuracy.

Multi-Scale inference We examine the contribution of multi-scale inference. As described in Section 4.2, RASST's retriever queries with speech windows at multiple scales and keeps the terms with the highest similarities. We compare against two variants. The first reuses our retriever but queries with only the largest window at inference time, but this introduces a training–inference mismatch, since the retriever was trained with MFA-aligned windows around each term. The second variant removes this mismatch by training the retriever directly with the largest window, without MFA-based window localization. Results are shown in Figure 8. Multi-scale inference combined with MFA-localized training performs best: it loses less than 1% in recall when scaling from 1K to 100K glossary, compared to drops of over 2% and 5% for the two variants.

Speech LLM Training Data We ablate the distractor-sampling strategy used to construct G_i during Speech LLM training. Besides retriever mined distractors in RASST, we train two variants: one with LLM-generated distractors, and one with no retrieved terms at all (equivalent to InfiniSST). At inference, all variants use the same retriever to populate G_i from the speech chunk before the Speech LLM generates the translation. Results are shown in Figure 12. The model trained with LLM-generated distractors achieves slightly lower terminology accuracy and BLEU than RASST. More surprisingly, *InfiniSST + RAG*, which sees no retrieved terms during training but is given them at inference, matches RASST's terminology accuracy but has much worse BLEU score. Inspecting its outputs, we find that it tends to copy retrieved terms verbatim without checking whether they actually appear in the source or inserting them at the right time.

More ablations on retriever training data, encoder choice and context length can be found in Appendix E.

7 Conclusion

We presented RASST, a retrieval-augmented framework for simultaneous speech translation that improves domain-specific terminology translation. RASST couples a lightweight cross-modal retriever with a Speech LLM and is trained with noise-robust supervision so that it remains accurate under realistic retrieval errors at inference time. On the ACL 60/60 dev set and the ESO test set, RASST improves terminology accuracy by nearly 40% and BLEU by up to 3 points over strong baselines.

Limitations

This work has several limitations. First, we focus exclusively on English as the source language. Whether RASST's cross-modal retriever and multi-scale inference transfer to other source languages remains untested. Second, our evaluation covers only two domains (academic talks and oncology training videos), leaving RASST's effectiveness on other terminology-rich settings such as legal and financial domains uncharacterized. Third, we rely on a fixed similarity threshold τ to suppress retrieval noise. Fourth, we do not explicitly model homophone confusion as in some prior contextual-biasing ASR work, since our focus is the integration of retrieval with an LLM-based simultaneous translation model, but it should be a natural extension. Finally, all experiments use Qwen3-Omni as the Speech LLM backbone, and we have not yet verified that our findings generalize to other speech LLMs.

References

Idris Abdulmumin, Victor Agostinelli, Tanel Alumäe, Antonios Anastasopoulos, Luisa Bentivogli, Ondřej Bojar, Claudia Borg, Fethi Bougares, Roldano Cattoni, Mauro Cettolo, Lizhong Chen, William Chen, Raj Dabre, Yannick Estève, Marcello Federico, Mark Fishel, Marco Gaido, Dávid Javorský, Marek Kasztelnik, and 33 others. 2025. Findings of the IWSLT 2025 evaluation campaign. In *Proceedings of the 22nd International Conference on Spoken Language Translation (IWSLT 2025)*, pages 412–481, Vienna, Austria (in-person and online). Association for Computational Linguistics.

Ibrahim Said Ahmad, Antonios Anastasopoulos, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, William Chen, Qianqian Dong, Marcello Federico, Barry Haddow, Dávid Javorský, Mateusz Krubiński, Tsz Kin Lam, Xutai Ma, Prashant Mathur, Evgeny Matusov, Chandresh Maurya, John P. McCrae, and 25 others. 2024. FINDINGS OF THE IWSLT 2024 EVALUATION CAMPAIGN. In *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*, pages 1–11, Bangkok, Thailand (in-person and online). Association for Computational Linguistics.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. 2021. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. In *Interspeech 2021*, pages 3670–3674.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Shanbo Cheng, Yu Bao, Zhichao Huang, Yu Lu, Ningxin Peng, Lu Xu, Runsheng Yu, Rong Cao, Yujiao Du, Ting Han, Yuxiang Hu, Zeyang Li, Sitong Liu, Shengtao Ma, Shiguang Pan, Jiongchen Xiao, Nuo Xu, Meng Yang, Rong Ye, and 9 others. 2025. Seed liveinterpret 2.0: End-to-end simultaneous speech-to-speech translation with your voice. *Preprint*, arXiv:2507.17527.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Xian Shi, Keyu An, and 1 others. 2025. Cosyvoice 3: Towards in-the-wild speech generation via scaling-up and post-training. *arXiv preprint arXiv:2505.17589*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Biao Fu, Donglei Yu, Minpeng Liao, Chengxi Li, Yidong Chen, Kai Fan, and Xiaodong Shi. 2025. Efficient and adaptive simultaneous speech translation with fully unidirectional architecture. *Preprint*, arXiv:2504.11809.

Li Fu, Yu Xin, Sunlu Zeng, Lu Fan, Youzheng Wu, and Xiaodong He. 2026. Pac: Pronunciation-aware contextualized large language model-based automatic speech recognition. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16377–16381.

Xun Gong, Anqi Lv, Zhiming Wang, and Yanmin Qian. 2024. Contextual Biasing Speech Recognition in Speech-enhanced Large Language Model. In *Interspeech 2024*, pages 257–261.

Xun Gong, Anqi Lv, Wangyou Zhang, Zhiming Wang, Huijia Zhu, and Yanmin Qian. 2025. BR-ASR: Efficient and Scalable Bias Retrieval Framework for Contextual Biasing ASR in Speech LLM. In *Interspeech 2025*, pages 4043–4047.

Shoutao Guo, Xiang Li, Mengge Liu, Wei Chen, and Yang Feng. 2025. Streamuni: Achieving streaming speech translation with a unified large speech-language model. *Preprint*, arXiv:2507.07803.

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*.

Christian Huber, Juan Hussain, Sebastian Stüker, and Alexander Waibel. 2021. Instant one-shot word-learning for context-specific neural sequence-to-sequence speech recognition. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–7.

Matthias Huck, Viktor Hangya, and Alexander Fraser. 2019. Better OOV translation with bilingual terminology mining. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5809–5815, Florence, Italy. Association for Computational Linguistics.

Jorge Iranzo-Sánchez, Jaume Santamaría-Jordà, Gerard Mas-Mollà, Gonçal V. Garcés Díaz-Munío, Javier Iranzo-Sánchez, Javier Jorge, Joan Albert Silvestre-Cerdà, Adrià Giménez, Jorge Civera, Albert Sanchis, and Alfons Juan. 2025. Speech translation for multilingual medical education leveraged by large language models. *Artificial Intelligence In Medicine*, 166:103147.

Mahaveer Jain, Gil Keren, Jay Mahadeokar, Geoffrey Zweig, Florian Metze, and Yatharth Saraf. 2020. Contextual RNN-T for Open Domain ASR. In *Interspeech 2020*, pages 11–15.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.

Sai Muralidhar Jayanthi, Devang Kulshreshtha, Saket Dingliwal, Srikanth Ronanki, and Sravan Bodapati. 2023. Retrieve and copy: Scaling ASR personalization to large catalogs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 631–639, Singapore. Association for Computational Linguistics.

Hong Jiang. 2013. The interpreters glossary in simultaneous interpreting: A survey. *Interpreting*, 15(1):74–93.

Qingnan Jiang, Mingxuan Wang, Jun Cao, Shanbo Cheng, Shujian Huang, and Lei Li. 2021. Learning kernel-smoothed machine translation with retrieved examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7280–7290, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. MetricX-24: The Google submission to the WMT 2024 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 492–504, Miami, Florida, USA. Association for Computational Linguistics.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. Nearest neighbor machine translation. In *International Conference on Learning Representations*.

Yuxiang Kong, Fan Cui, Liyong Guo, Heinrich Dinkel, Lichun Fan, Junbo Zhang, and Jian Luan. 2025. GLCLAP: A Novel Contrastive Learning Pre-trained Model for Contextual Biasing in ASR. In *Interspeech 2025*, pages 5173–5177.

Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024. TransLLaMa: LLM-based simultaneous translation system. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 461–476, Miami, Florida, USA. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Duc Le, Mahaveer Jain, Gil Keren, Suyoun Kim, Yangyang Shi, Jay Mahadeokar, Julian Chan, Yuan Shangguan, Christian Fuegen, Ozlem Kalinli, Yatharth Saraf, and Michael L. Seltzer. 2021a. Contextualized Streaming End-to-End Speech Recognition with Trie-Based Deep Biasing and Shallow Fusion. In *Interspeech 2021*, pages 1772–1776.

Duc Le, Gil Keren, Julian Chan, Jay Mahadeokar, Christian Fuegen, and Michael L. Seltzer. 2021b. Deep shallow fusion for rnn-t personalization. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 251–257.

Bryan Li, Jiaming Luo, Eleftheria Briakou, and Colin Cherry. 2025. Leveraging domain knowledge at inference time for LLM translation: Retrieval versus generation. In *Proceedings of the 4th International Workshop on Knowledge-Augmented Methods for Natural Language Processing*, pages 91–106, Albuquerque, New Mexico, USA. Association for Computational Linguistics.

Christopher Li, Pat Rondon, Diamantino Caseiro, Leonid Velikovich, Xavier Velez, and Petar Aleksic. 2021. Improving entity recall in automatic speech recognition with neural embeddings. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7353–7357.

Siqi Li, Danni Liu, and Jan Niehues. 2024. Optimizing rare word accuracy in direct speech translation with a retrieval-and-demonstration approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12703–12719, Miami, Florida, USA. Association for Computational Linguistics.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. 2020a. SIMULEVAL: An evaluation toolkit for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 144–150, Online. Association for Computational Linguistics.

Xutai Ma, Juan Pino, and Philipp Koehn. 2020b. SimulMT to SimulST: Adapting simultaneous text translation to end-to-end simultaneous speech translation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 582–587, Suzhou, China. Association for Computational Linguistics.

Yuxian Meng, Xiaoya Li, Xiayu Zheng, Fei Wu, Xiaofei Sun, Tianwei Zhang, and Jiwei Li. 2022. Fast nearest neighbor machine translation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 555–565, Dublin, Ireland. Association for Computational Linguistics.

Tsendsuren Munkhdalai, Khe Chai Sim, Angad Chandorkar, Fan Gao, Mason Chua, Trevor Strohman, and Françoise Beaufays. 2022. Fast contextual adaptation with neural associative memory for on-device personalized speech recognition. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6632–6636.

Tsendsuren Munkhdalai, Zelin Wu, Golan Pundak, Khe Chai Sim, Jiayang Li, Pat Rondon, and Tara N. Sainath. 2023. Nam+: Towards scalable end-to-end contextual biasing for adaptive asr. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 190–196.

Siqi Ouyang, Xi Xu, and Lei Li. 2025a. CMU's IWSLT 2025 simultaneous speech translation system. In *Proceedings of the 22nd International Conference on Spoken Language Translation (IWSLT 2025)*, pages 309–314, Vienna, Austria (in-person and online). Association for Computational Linguistics.

Siqi Ouyang, Xi Xu, and Lei Li. 2025b. InfiniSST: Simultaneous translation of unbounded speech with large language model. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3032–3046, Vienna, Austria. Association for Computational Linguistics.

Sara Papi, Marco Gaido, Matteo Negri, and Luisa Bentivogli. 2024. StreamAtt: Direct streaming speech-to-text translation with attention-based audio history selection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3692–3707, Bangkok, Thailand. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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.

Golan Pundak, Tara N. Sainath, Rohit Prabhavalkar, Anjuli Kannan, and Ding Zhao. 2018. Deep context: End-to-end contextual speech recognition. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 418–425.

Yi Ren, Jinglin Liu, Xu Tan, Chen Zhang, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2020. SimulSpeech: End-to-end simultaneous speech to text translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3787–3796, Online. Association for Computational Linguistics.

Elizabeth Salesky, Kareem Darwish, Mohamed Al-Badrashiny, Mona Diab, and Jan Niehues. 2023. Evaluating multilingual speech translation under realistic conditions with resegmentation and terminology. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 62–78, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.

Minghan Wang, Thuy-Trang Vu, Yuxia Wang, Ehsan Shareghi, and Gholamreza Haffari. 2025. Conversational SimulMT: Efficient simultaneous translation with large language models. In *Proceedings of the 22nd International Conference on Spoken Language Translation (IWSLT 2025)*, pages 93–105, Vienna, Austria (in-person and online). Association for Computational Linguistics.

Suhang Wu, Jialong Tang, Chengyi Yang, Pei Zhang, Baosong Yang, Junhui Li, Junfeng Yao, Min Zhang, and Jinsong Su. 2025. Locate-and-focus: Enhancing terminology translation in speech language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11345–11360, Vienna, Austria. Association for Computational Linguistics.

Zelin Wu, Tsendsuren Munkhdalai, Pat Rondon, Golan Pundak, Khe Chai Sim, and Christopher Li. 2023. Dual-Mode NAM: Effective Top-K Context Injection for End-to-End ASR. In *Interspeech 2023*, pages 221–225.

Zelin Wu, Gan Song, Christopher Li, Pat Rondon, Zhong Meng, Xavier Velez, Weiran Wang, Diamantino Caseiro, Golan Pundak, Tsendsuren Munkhdalai, Angad Chandorkar, and Rohit Prabhavalkar. 2024. Deferred NAM: Low-latency top-k context injection via deferred context encoding for non-streaming ASR. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 315–323, Mexico City, Mexico. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Guanrou Yang, Ziyang Ma, Zhifu Gao, Shiliang Zhang, and Xie Chen. 2024. Ctc-assisted llm-based contextual asr. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 126–131.

Bolaji Yusuf, Aditya Gourav, Ankur Gandhe, and Ivan Bulyko. 2023. On-the-fly text retrieval for end-to-end asr adaptation. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Zixin Zeng, Rui Wang, Yichong Leng, Junliang Guo, Shufang Xie, Xu Tan, Tao Qin, and Tie-Yan Liu. 2023. Extract and attend: Improving entity translation in neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1697–1710, Toronto, Canada. Association for Computational Linguistics.

Jingyi Zhang, Masao Utiyama, Eiichro Sumita, Graham Neubig, and Satoshi Nakamura. 2018. Guiding neural machine translation with retrieved translation pieces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1325–1335, New Orleans, Louisiana. Association for Computational Linguistics.

Tong Zhang, Long Zhang, Wei Ye, Bo Li, Jinan Sun, Xiaoyu Zhu, Wen Zhao, and Shikun Zhang. 2021. Point, disambiguate and copy: Incorporating bilingual dictionaries for neural machine translation. 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 3970–3979, Online. Association for Computational Linguistics.

Ding Zhao, Tara N. Sainath, David Rybach, Pat Rondon, Deepti Bhatia, Bo Li, and Ruoming Pang. 2019. Shallow-Fusion End-to-End Contextual Biasing. In *Interspeech 2019*, pages 1418–1422.

Wenhao Zhu, Jingjing Xu, Shujian Huang, Lingpeng Kong, and Jiajun Chen. 2023. INK: Injecting kNN knowledge in nearest neighbor machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15948–15959, Toronto, Canada. Association for Computational Linguistics.

A Wikidata Processing

We normalize each label by removing parenthetical qualifiers, lowercasing for deduplication, and retaining only short alphabetic or hyphenated terms with at most five words. We further require a non-empty short description from the glossary index that does not simply repeat the term, which filters out many ambiguous or non-English labels. To avoid a corpus dominated by very frequent instance types such as people, taxa, or scholarly articles, we assign each entity a primary `P31` type using its least frequent instance type and rank terms with inverse-type-frequency sampling. Terms appearing in the held-out evaluation glossaries are removed before training.

For each retained term, we prompt Gemini-2.5-Flash to generate three natural utterances of 5–15 words containing the term (Figure 14). We then synthesize the utterances with CosyVoice 3[11], using speaker prompts randomly sampled from GigaSpeech.

B Speech LLM Data Synthesis

As in InfiniSST, we build robust segments of 28.8 seconds, filter out those with WER larger than 0.1 using parakeet-0.6b-tdt-v2[12] and generate the target language translation with Qwen3-30B-A3B-

[11] https://huggingface.co/FunAudioLLM/Fun-CosyVoice3-0.5B-2512

[12] https://huggingface.co/nvidia/parakeet-tdt-0.6b-v2

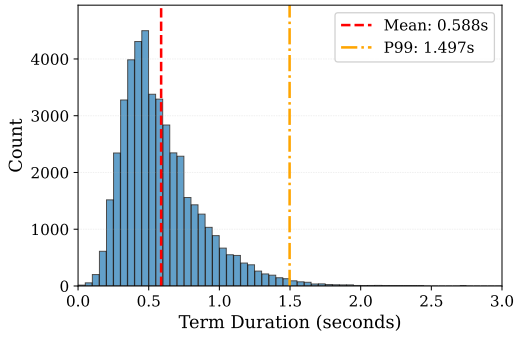

Figure 6: The distribution of term duration on GigaSpeech training set.

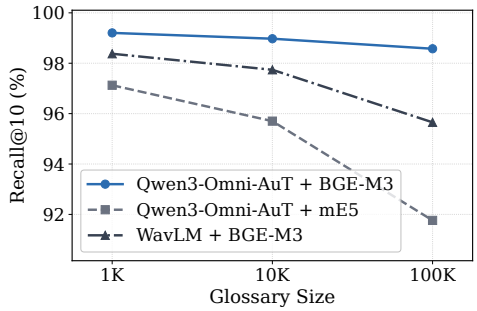

Figure 7: Recall@10 across glossary sizes for different encoder combinations. Qwen3-Omni-AuT + BGE-M3 achieves the highest recall.

Instruct-2507-FP8 (Yang et al., 2025)[13] and filter out those with MetricX score (Juraska et al., 2024)[14] worse than -3. Finally we build the interleaved training data using SimAlign (Jalili Sabet et al., 2020) with LaBSE (Feng et al., 2022).

Translation Prompt

```
You are given English document split into
lines. Translate each line into Chinese. Do
not include any other text.
<begin>
{}
<end>
```

C Term Duration Distribution

We show the distribution of term duration on GigaSpeech training set in Figure 6.

D Training Details

Retriever We train the retriever with LoRA applied to both the speech and text encoders. For the audio encoder, LoRA is inserted into the query/key/value/output projections and MLP blocks with rank r=32 and scaling factor α=64. For the text encoder, LoRA targets the query/key/value projections with rank r=16 and α=32. We train with a multi-positive InfoNCE objective using a fixed temperature τ=0.03. We optimize with AdamW (learning rate 1×10^{-4}, weight decay 0.01), cosine annealing with 10% warmup, and gradient clipping (max norm 1.0) under BF16 mixed precision. Training is performed with Distributed Data Parallel (DDP) on eight RTX A6000 GPUs with a global batch size of 8192.

Speech LLM We fine-tune the Speech LLM using LoRA with rank r=8 and α=32. We enable sequence packing with a maximum sequence length of 2028 tokens. We optimize with AdamW (learning rate 1×10^{-4}, weight decay 0.01) and gradient

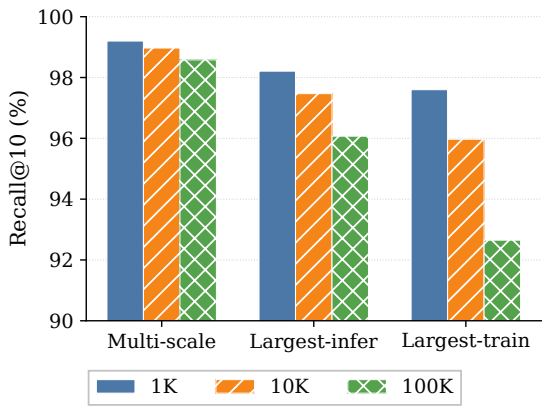

Figure 8: Recall@10 on the GigaSpeech dev set for multi-scale inference in RASST and two variants. Largest-infer uses the same retriever as RASST but queries with only the largest window at inference. Largest-train trains the retriever with the largest window instead of the MFA-localized window. Multi-scale inference combined with MFA-localized training performs best.

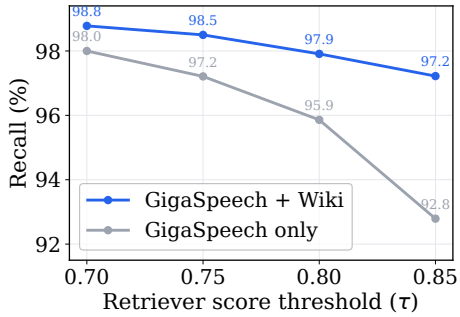

Figure 9: Recall@10 under different threshold τ on the GigaSpeech dev set for retrievers trained with and without Wikidata.

[13] https://huggingface.co/Qwen/Qwen3-30B-A3B-Instruct-2507-FP8

[14] https://huggingface.co/google/metricx-24-hybrid-xl-v2p6

Context Length	1K	10K	100K
Fixed 1.92s	98.11	97.47	95.18
Fixed 3.84s	98.65	98.31	97.58
Fixed 5.76s	**99.64**	**99.58**	**99.33**
Variable 2.88–5.76s	99.20	98.97	98.58

Table 3: Recall@10 of retrievers trained with different context lengths, across glossary sizes. Longer speech contexts yield higher recall.

clipping (max norm 1.0), with a global batch size of 4. Training is performed with DDP and expert parallelism (EP) of size 2 on two L40S GPUs.

E Additional Ablations

Additional ablations that cannot fit in the main text is here.

Retriever training data We examine the impact of training data by comparing our default retriever, trained on both GigaSpeech and Wikidata, against a GigaSpeech-only variant. As shown in Figure 9, adding Wikidata substantially improves recall, with the largest gains at high τ, indicating that the retriever assigns higher similarity to true positives when trained with Wikidata.

Retriever encoders We test different encoder combinations here. We additionally tried mE5 text encoder (Wang et al., 2024)[15] and WavLM-large speech encoder (Chen et al., 2022)[16]. As shown in Figure 7, Qwen3-Omni-Aut + BGE-M3 achieves the best recall.

Retriever context length. RASST trains the retriever with variable-length speech contexts and here we compare against fixed-length alternatives. As shown in Table 3, longer fixed-length contexts generally yield higher recall across glossary sizes. The variable-length setup matches the recall of the longest fixed-length context (5.76s) while offering better flexibility at inference.

[15] https://huggingface.co/intfloat/multilingual-e5-large

[16] https://huggingface.co/microsoft/wavlm-large

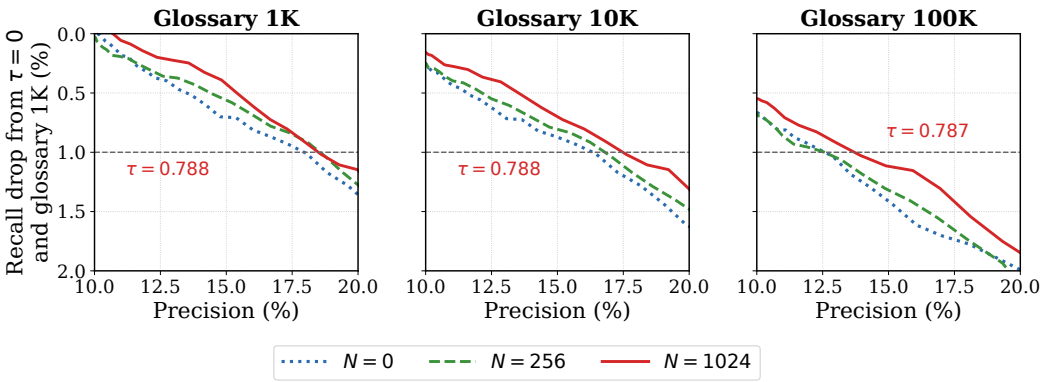

Figure 10: Precision–recall curves across glossary sizes for retrievers trained with N=0, 256, 1024 hard negatives, obtained by sweeping $\tau \in [0, 1]$. The dashed vertical line marks τ=0.78, the operating point at which recall drops by 1% from τ=0 at the 1K glossary.

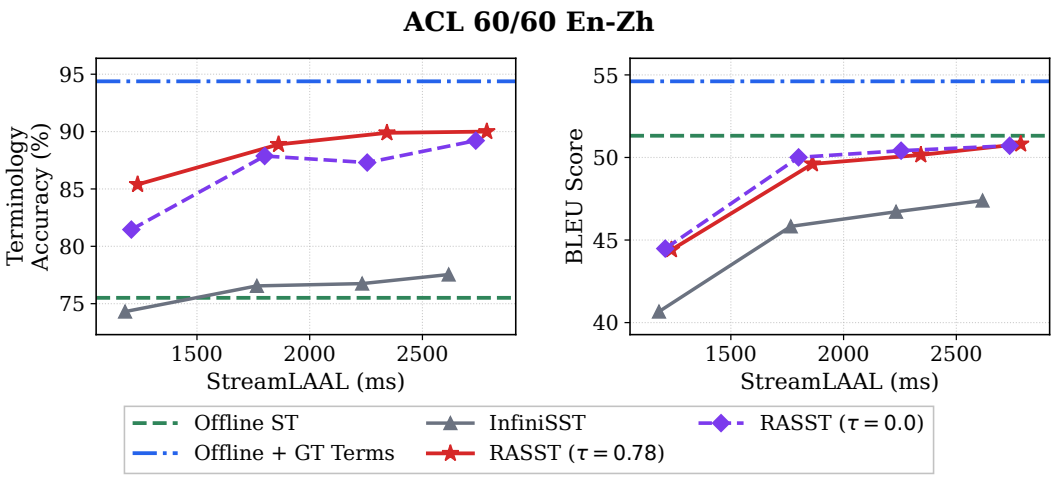

Figure 11: Quality–latency trade-off of RASST with $\tau = 0$ and $\tau = 0.78$.

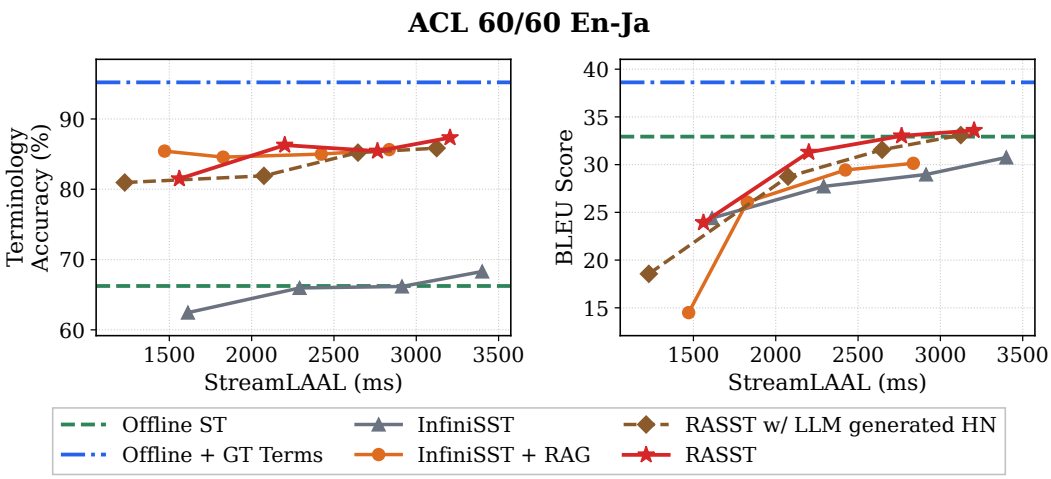

Figure 12: Quality–latency trade-off of RASST against two training-data variants: a version trained with LLM-generated distractors, and *InfiniSST + RAG*, which adds retrieval at inference to a model not trained with terms G_i. RASST achieves the best overall trade-off.

System	You are a professional simultaneous interpreter. You will be given chunks of English audio and you need to translate the audio into Chinese text. Use the term_map as a reference for terminology if provided.
User 1	<audio> **term_map:** things → 事; everything → 所有努力; right → 权利; thing → 东西; that thing → 那东西.
Assistant 1	她会做些我本意不想让她做的<term>事</term>。这样一来，之前<term>所有努力</term>都白费了，我不得不重新写一遍。
User 2	<audio> **term_map:** the Porch Gable → 门廊小屋; story → 传说; Anne → 安妮; seclusion → 静谧; Diana → 戴安娜; gabelle → 法国盐税; that story → 这个故事; porch → 门廊.
Assistant 2	然而，最终<term>传说</term>终于完成了，<term>安妮</term>在<term>门廊小屋</term>的<term>静谧</term>中把它读给了<term>戴安娜</term>听。她
User 3	<audio> **term_map:** Diana → 戴安娜; watchful eye → 密切留意; Robert Ray → 罗伯特·雷; her pathetic scene → 那个悲情场景; scene → 现场; that scene → 那场戏.
Assistant 3	成功地呈现了那个悲情场景，同时又没有牺牲<term>罗伯特·雷</term>，她一边读一边<term>密切留意</term>着<term>戴安娜</term>的反应。戴安娜

Figure 13: An example of En-Zh Speech LLM training data.

Prompt template for Wikidata term utterance generation

System instruction:
You are a dataset builder for speech recognition research.
Your task: given a list of terms with their short descriptions, generate short English utterances (5-15 words each) that a speaker might say in a lecture, tutorial, or technical discussion. Use the description as context to produce accurate, meaningful sentences, but do NOT just copy the description.
Each utterance MUST contain the given term EXACTLY as written (same spelling, but case may differ).
Vary sentence structure and term position (beginning, middle, end). Do NOT repeat the same pattern across terms.
Output valid JSON only - no markdown fences, no explanation.

User prompt:
Generate {variants_per_term} short utterances (5-15 words) for EACH of the following terms. Use the description as context to make the utterance accurate and meaningful. Each utterance must contain the term verbatim.

Terms with descriptions:
```
[
  {"term": "Seismology", "description": "scientific study of earthquakes"},
  {"term": "dataflow graph", "description": "graph representation of data dependencies"}
]
```

Return a JSON object mapping each term to a list of {variants_per_term} utterance strings. Example format:
```
{
  "Seismology": [
    "Seismology helps us understand earthquake patterns.",
    "The field of seismology has advanced rapidly."
  ]
}
```

IMPORTANT: output raw JSON only, no markdown code fences.

Figure 14: Prompt template for Wikidata term utterance generation.