Real-time Simultaneous Translation of Unbounded Streaming Speech

Lei Li

Language Technologies Institute | Carnegie Mellon University School of Computer Science

June 19, 2025


MARCO POLO
Travels between 1271 and 1295
Venice
Constantinople (Istanbul)
Bulgar
SIBERIA
Mediterranean Sea
Black Sea
Sarai
Aral Sea
Caspian Sea
Ayas
Erzurum
Acre
Jerusalem
Tabriz
Samarkand
Bukhara
HINDU KUSH MTS.
PAMIRS
Kashgar
TAKLA MAKAN DESERT
Khotan (Hotan)
IRAN
Persian Gulf
ARABIA
AFGHANISTAN
Hormuz
HIMALAYAS
GOBI DESERT
Suzhou
Lanzhou
Ningxia
Shangdu
JAPAN
Dadu (Beijing)
GANSU
CHINA
YELLOW SEA
Quinsay (Hangzhou)
Quanzhou
PACIFIC OCEAN
Yunnan
Tagaung
INDIA
SOUTH CHINA SEA
CHAMPA (VIETNAM)
ETHIOPIA
ARABIAN SEA
Nicobar Islands
Malay Peninsula
Ceylon (Sri Lanka)
INDIAN OCEAN
Sumatra
0 500 1000 mi
0 750 1500 km
© Encyclopædia Britannica, Inc.

Breaking Language Barriers

Cultural Communication

Education

Medical care

Tourism

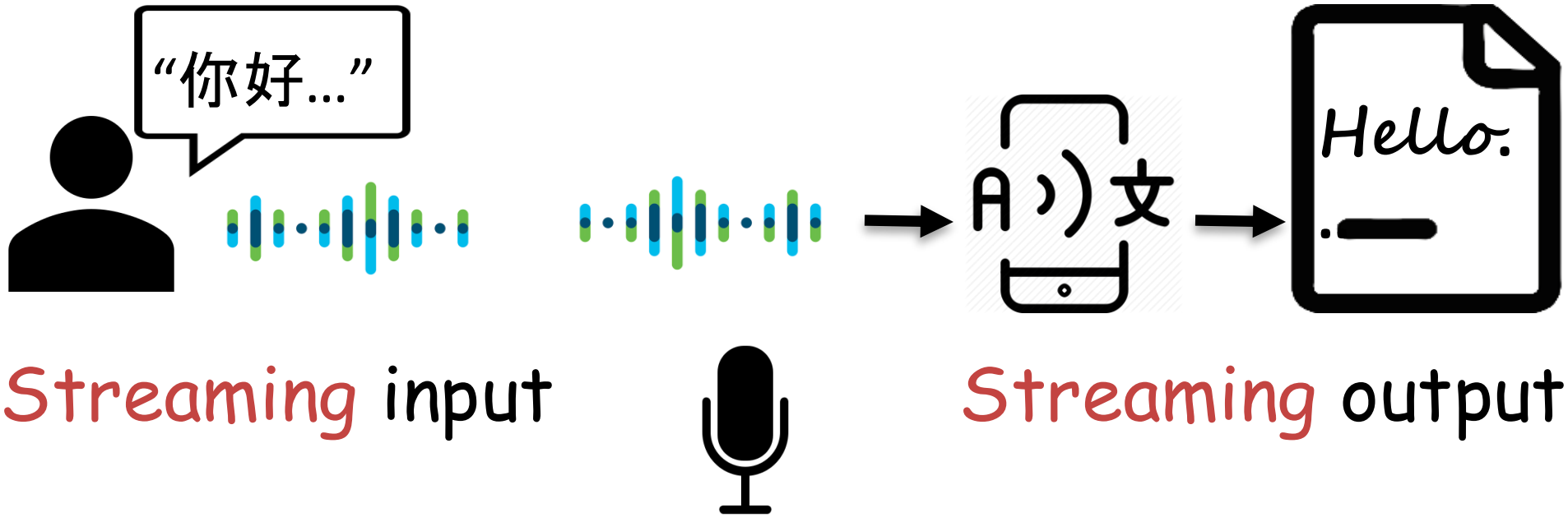
Business&trade

Outline

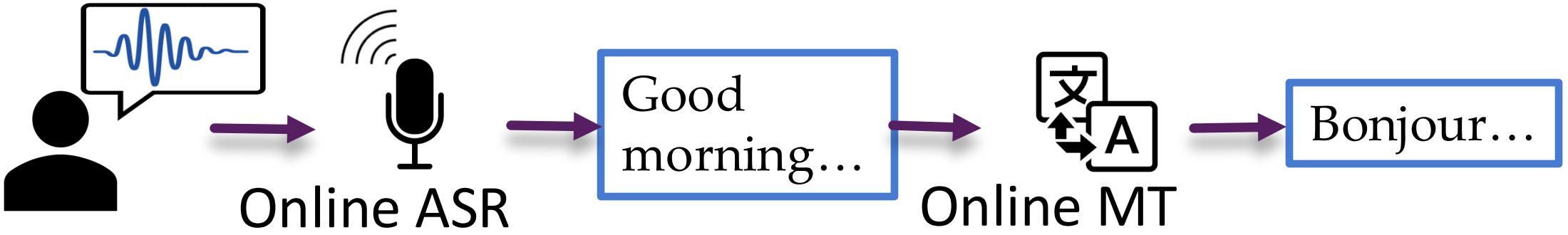
- Simultaneous Speech Translation (SST) and challenges
- InfiniSST: high-quality low-latency unbounded SST
  - Model design
  - Training data construction
  - Inference on unbounded SST
  - Experiment Evaluation

Simultaneous Speech-to-text Translation

- Read the audio signals of speech in one language, and translate to the text in another language while speaker speaks (SST).



Traditional Cascaded SST System



- Drawbacks:

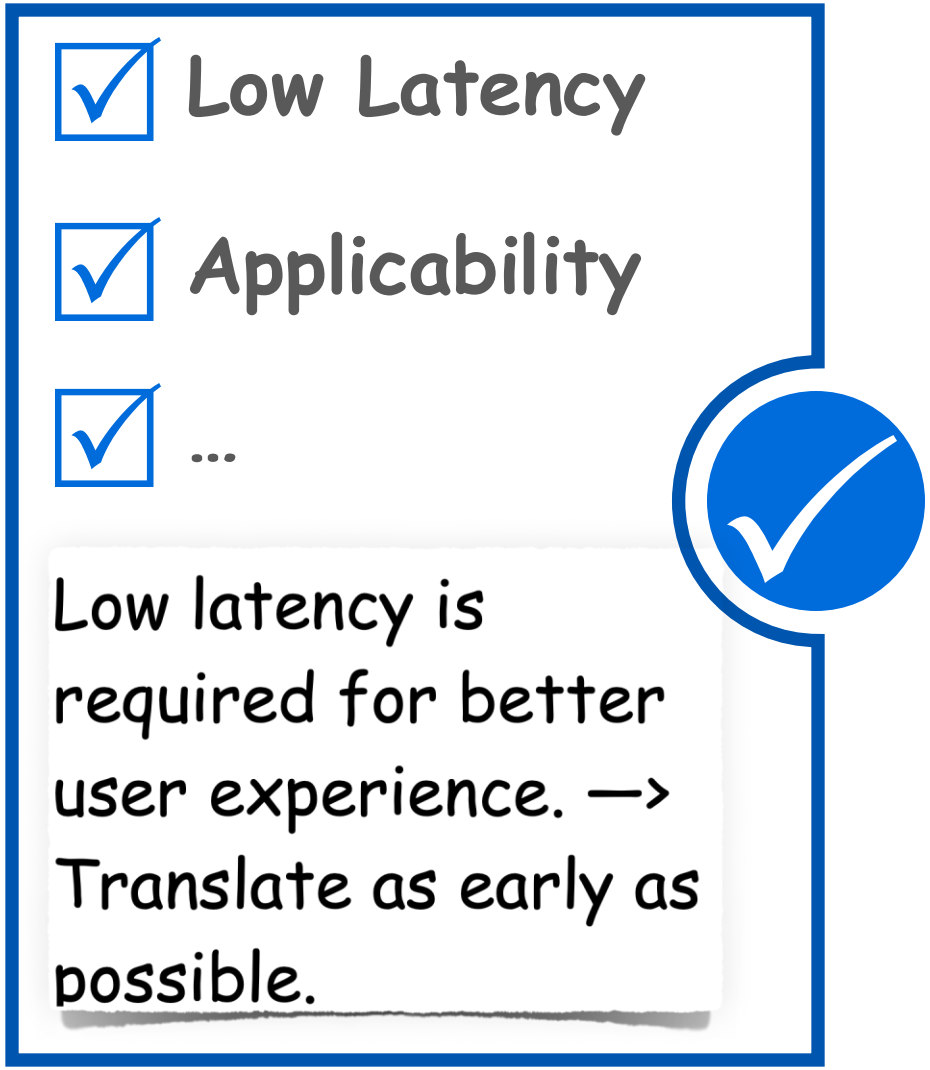
1. Computationally inefficient
2. Error propagation:

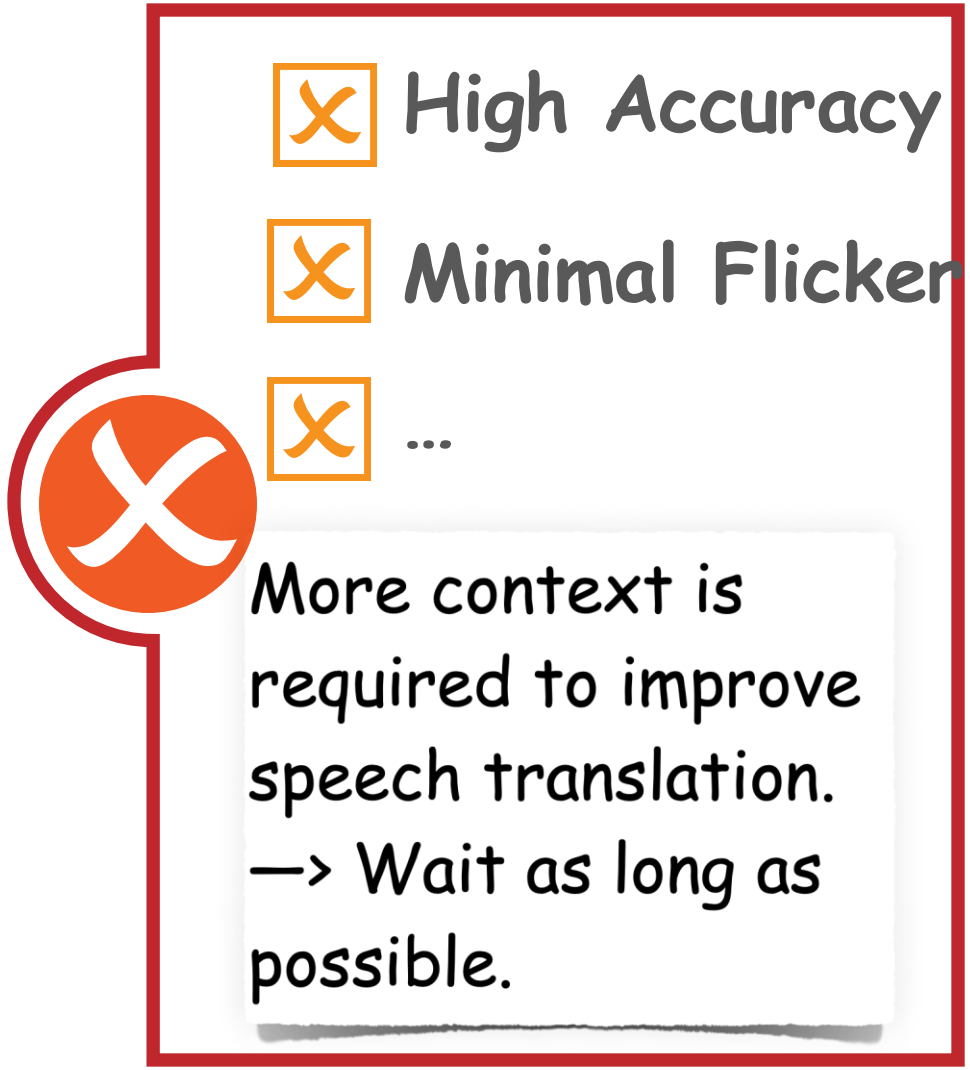
Wrong/error transcript recognition → Wrong translation

End-to-end SiST

- Goal: End2end streaming ST needs to balance the latency and quality, and generate translations based on the partial speech chunk with a single model.

Challenges for SST





Challenges of Unbounded Speech

- The audio is looooooooooong!
  - e.g. 1 hour talk
  - Out of memory (OOM)
  - Out of training distribution (OOD)
- How to avoid OOM and OOD, while achieving good trade-off between the translation quality and the system latency?

Prior Works

- Most of prior SST works are on segmented speech, usually less than 30 seconds, not directly applicable to unbounded speech.
  - MoSST: Learning When to Translate for Streaming Speech, ACL 2022
- StreamAtt is the only open-sourced one working on unbounded speech, but it is not computationally efficient
  - It preserves recent speech and generated translations.
  - Every step, the features of preserved speech and translation are recomputed. **THIS STEP is COSTLY.**

Outline

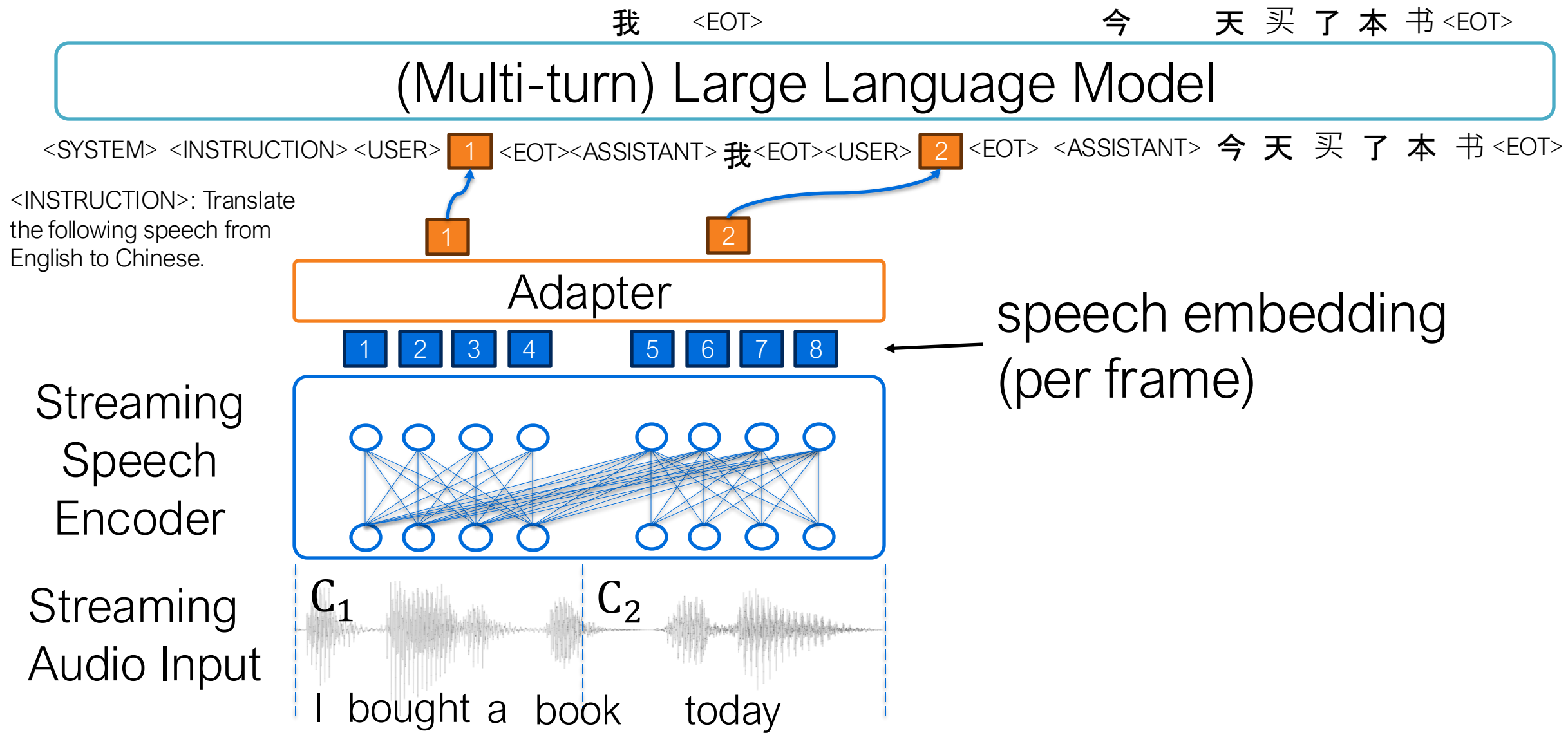
- Simultaneous Speech Translation (SST) and challenges
- InfiniSST: high-quality low-latency unbounded SST
  - Model design
  - Training data construction
  - Inference on unbounded SST
  - Experiment Evaluation

Introducing InfiniSST – Key Idea

- Ensure translation quality:
  - pre-trained speech encoder + LLM
- Reducing latency:
  - avoid recomputation ➔ incremental computation
  - Chat-style interleaving read/write policy
- Enable unbounded speech:
  - techniques to enable long context

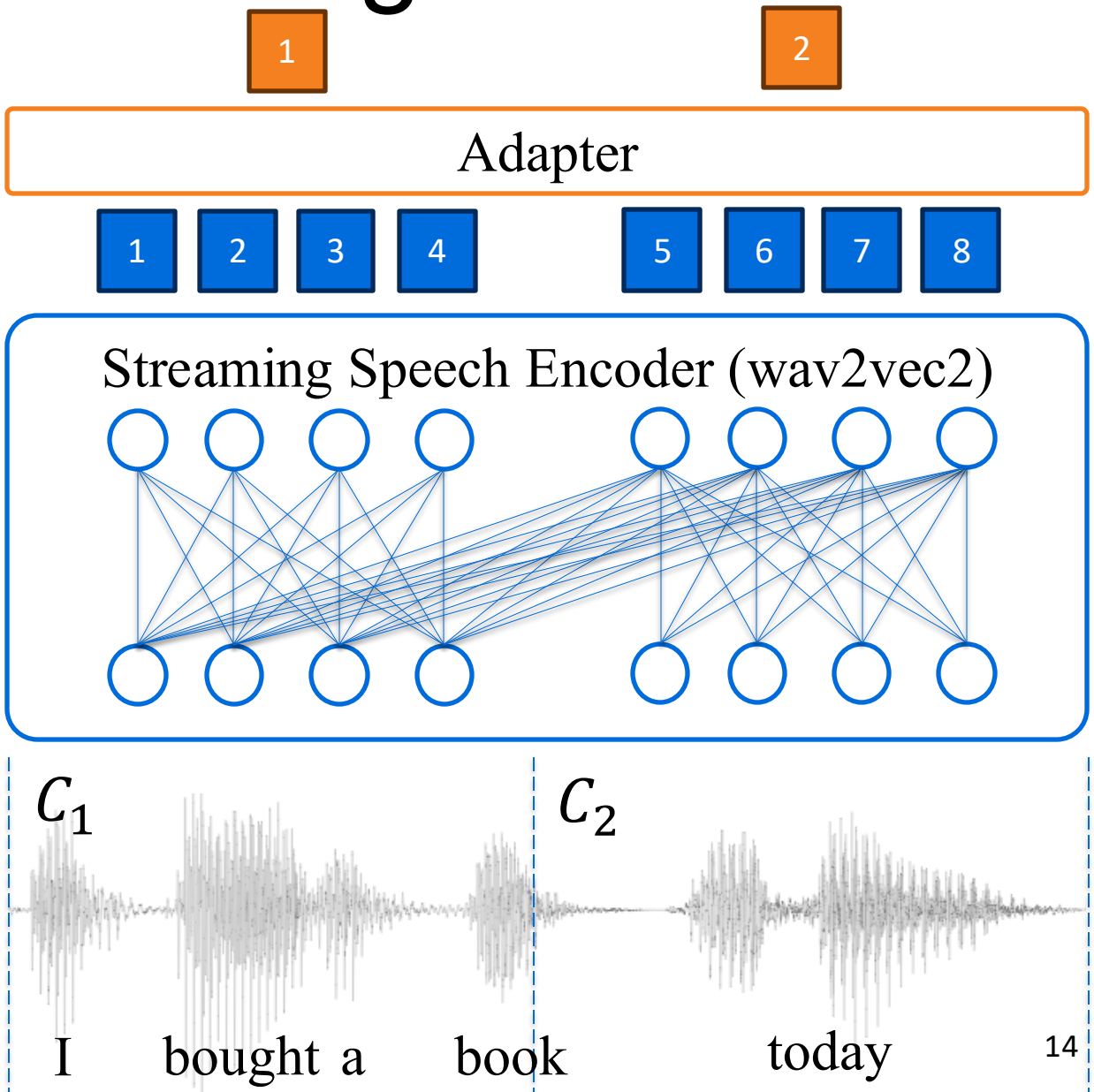
Siqi Ouyang, Xi Xu, Lei Li. InfiniSST: Simultaneous Translation of Unbounded Speech with Large Language Model. ACL 2025.

InfiniSST

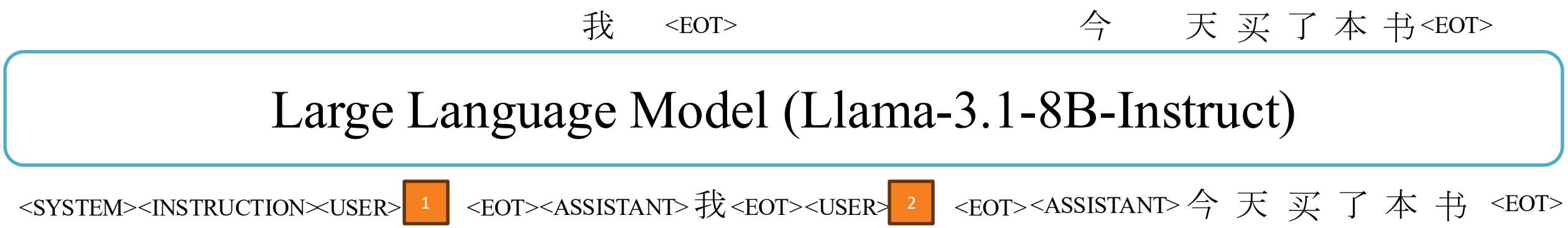

我 <EOT> 今 天 买 了 本 书 <EOT>
(Multi-turn) Large Language Model
<SYSTEM> <INSTRUCTION> <USER> 1 <EOT><ASSISTANT> 我<EOT><USER> 2 <EOT> <ASSISTANT> 今 天 买 了 本 书 <EOT>
<INSTRUCTION>: Translate the following speech from English to Chinese.
1
2
Adapter
1 2 3 4 5 6 7 8
speech embedding (per frame)
Streaming Speech Encoder
Streaming Audio Input
c_1
c_2
I bought a book today

Speech Encoding

- Speech chunk: 960ms (48x20ms)
- Chunkwise-causal encoder
  - Bidirectional inside chunk
  - Causal between chunks
  - Sliding window w^s
  - Rotary position embedding
- Speech-to-Token Embedding Adapter
  - Map to LLM embedding space
  - Shrink length by 4



Multi-turn LLM Decoding



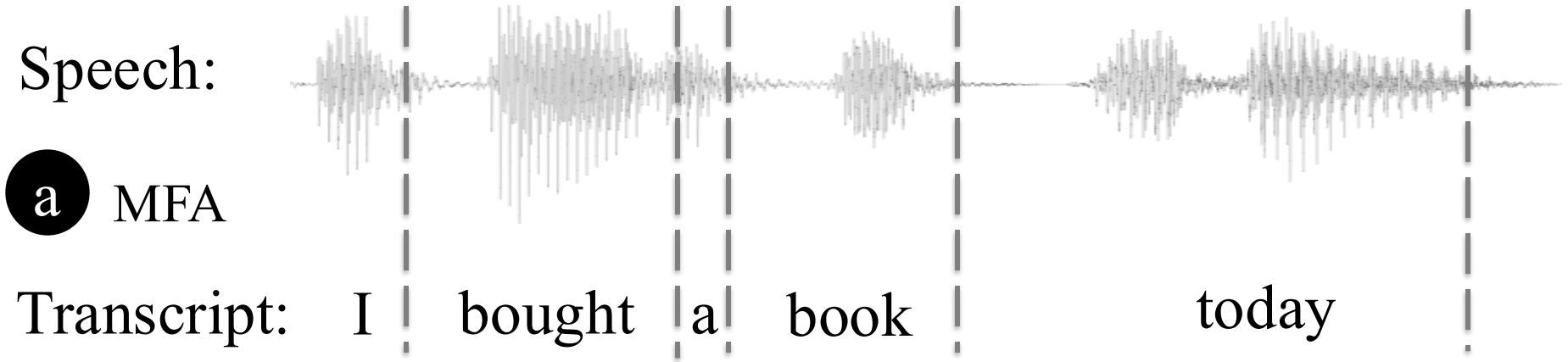
- Multi-turn dialogue format:
- Instruction: Translate the following speech from <LangX> to <LangY>.
- LLM stops the current turn of translation at <EOT>

Training Data Construction

- MuST-C: triplets of <speech, transcript, translation>.
  - Each triplet is a segmented utterance from a complete TED Talk.
- Data trajectories for training:
  - Trajectory is an action sequence (s1, t1, s2, t2, …) alternating between speech reading and translation writing.
  - Each speech reading is of duration 960 ms
  - Each translation writing ends with <EOT>

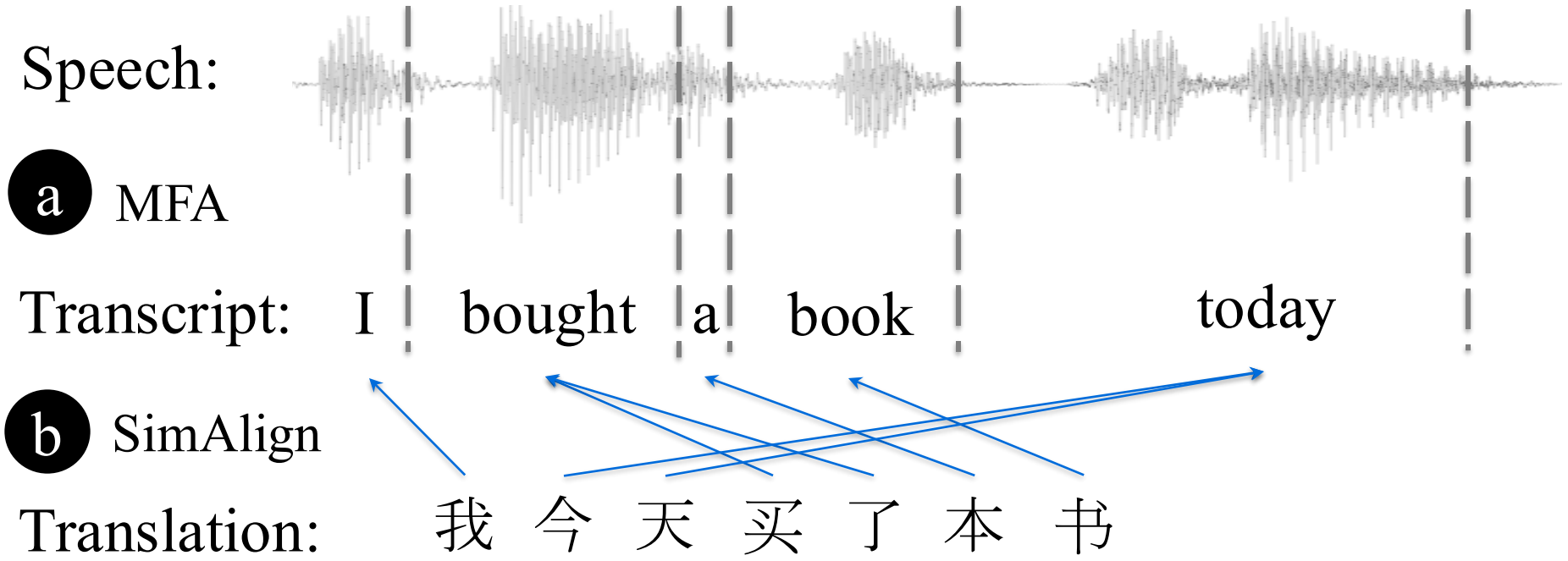
Speech-Text Trajectory Construction

Aligning speech frames with transcript tokens



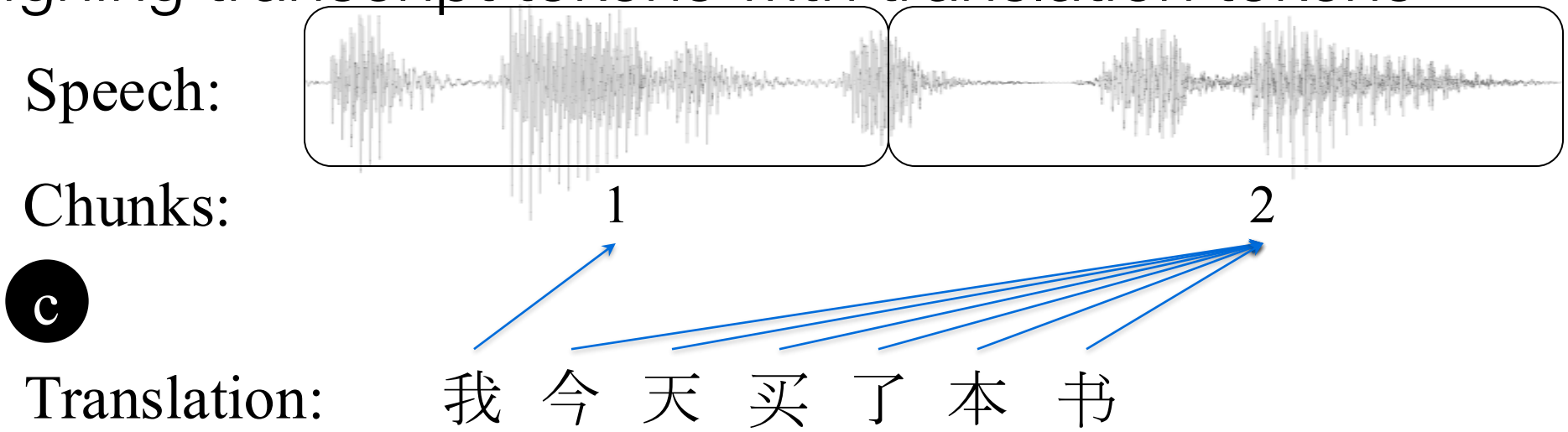
Speech-Text Trajectory Construction

Aligning speech frames with transcript tokens
Aligning transcript tokens with translation tokens



Speech-Text Trajectory Construction

Aligning speech frames with transcript tokens
Aligning transcript tokens with translation tokens



Monotonically grouped by speech chunks

Data Construction: Robust Segments

- Segmented speech utterances primarily consist of human speech; however, non-linguistic sounds (e.g., laughter, applause) are also present.

- To enhance the robustness of the SST dataset, we cut the entire TED Talk evenly into robust segments that each span 30 speech chunks, i.e., 28.8 seconds.

Data Construction: Multi-Latency Augment

- The trajectory we just built might be “too perfect”.
- We randomly select $m \in [1, M]$, so that every m neighbouring steps of a trajectory is merged together.
  - Given m=2, a trajectory (s1, t1, s2, t2, s3, t3, s4, t4) becomes (s1+s2, t1+t2, s3+s4, t3+t4)
- This constructs trajectories with larger latency.

InfiniSST Training

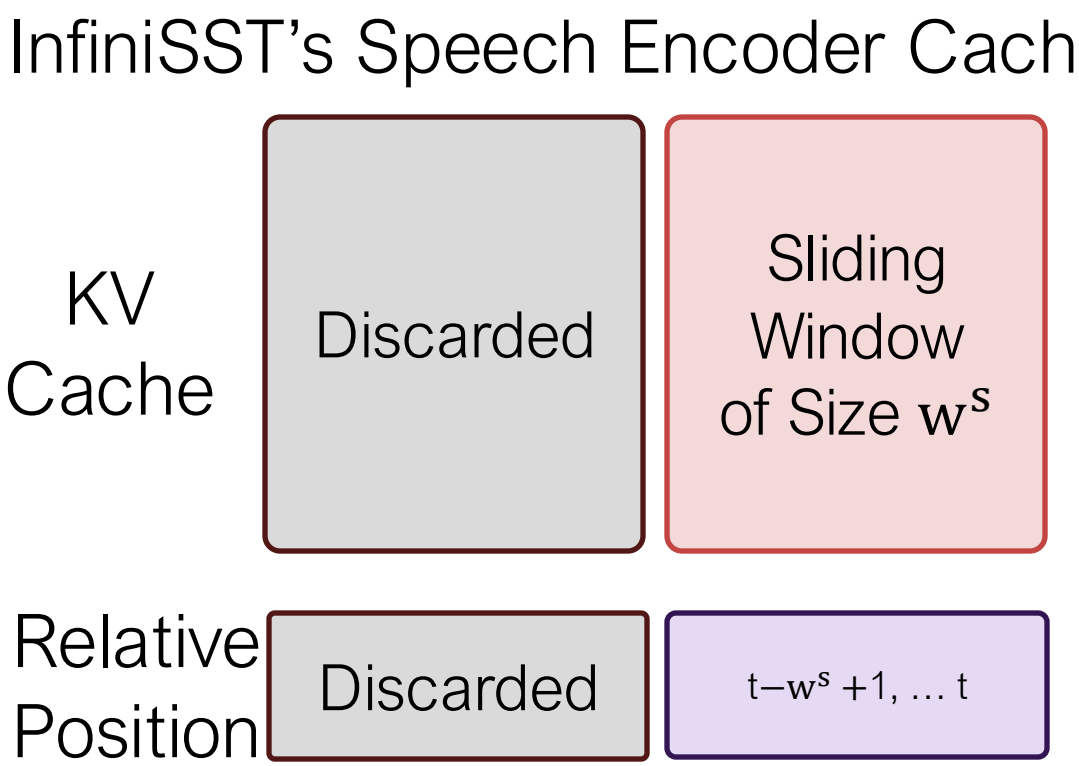
- Train InfiniSST with multi-latency augmented trajectories from robust segments of MuST-C dataset.
- Two-stage training
  - Freeze LLM, finetune speech encoder and adapter
  - Freeze speech encoder and adapter, finetune LLM
- Loss only applied to translation entries of trajectory, including <EOT> tokens.

Inference on Unbounded Speech

- Unbounded speech input are cut into chunks of 960ms.
- Latency multiplier m is selected.
- Perform inference after every m chunks are received.

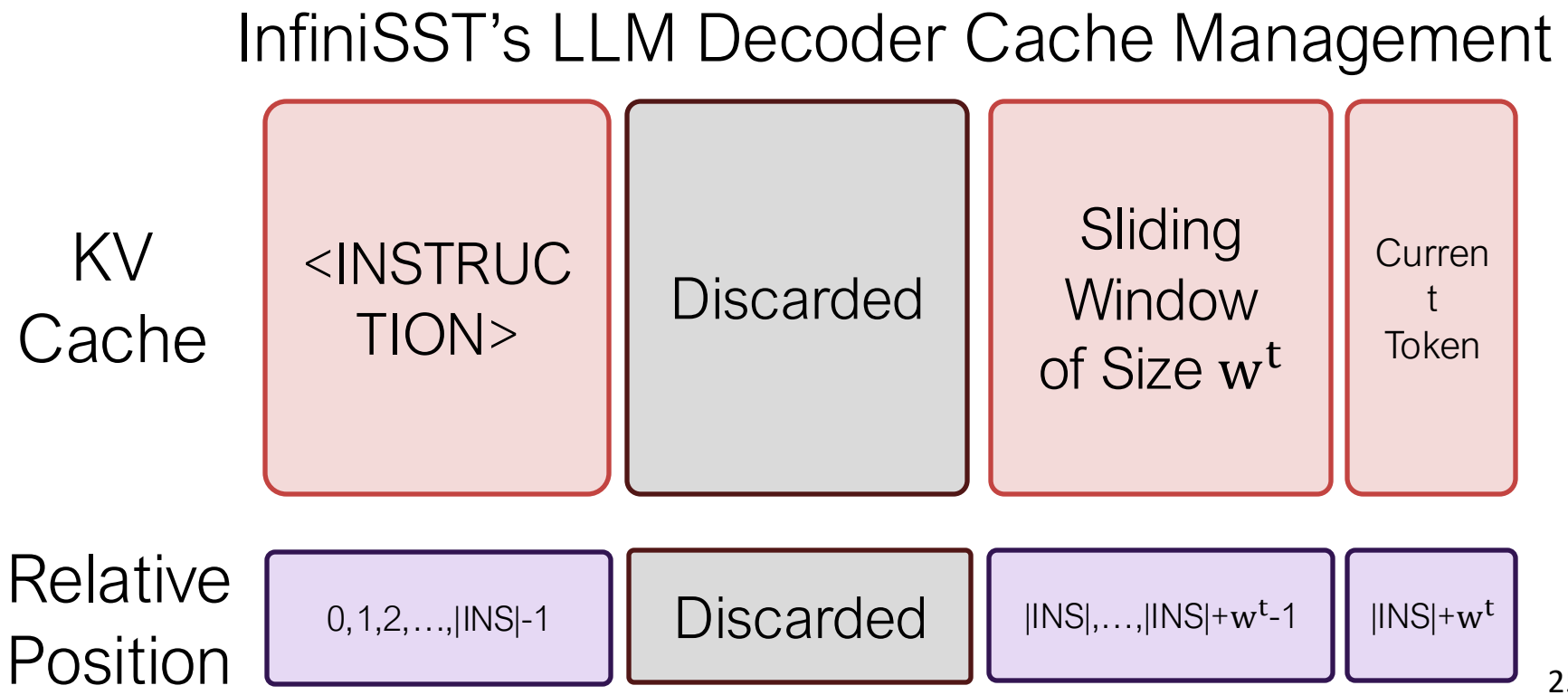
Inference on Unbounded Speech

- Both LLM and speech encoder maintain KV cache before RoPE.
- Speech encoder keeps KV cache using the sliding window mechanism (size=10)



Inference on Unbounded Speech

- LLM and speech encoder maintain KV cache before RoPE.
- At step i, we receive chunks $im, im+1, \ldots, (i+1)m-1$
- win size=1000



Outline

- Simultaneous Speech Translation (SST) and challenges
- InfiniSST: high-quality low-latency unbounded SST
  - Model design
  - Training data construction
  - Inference on unbounded SST
  - Experiment Evaluation

Dataset

- MuST-C
  - Languages: En-Es, En-De, En-Zh
  - Training: ~400 hours each
- Data filtering for En-Zh
  - CometKiwi + TowerInstruct
- Trajectory and robust segment construction as mentioned before

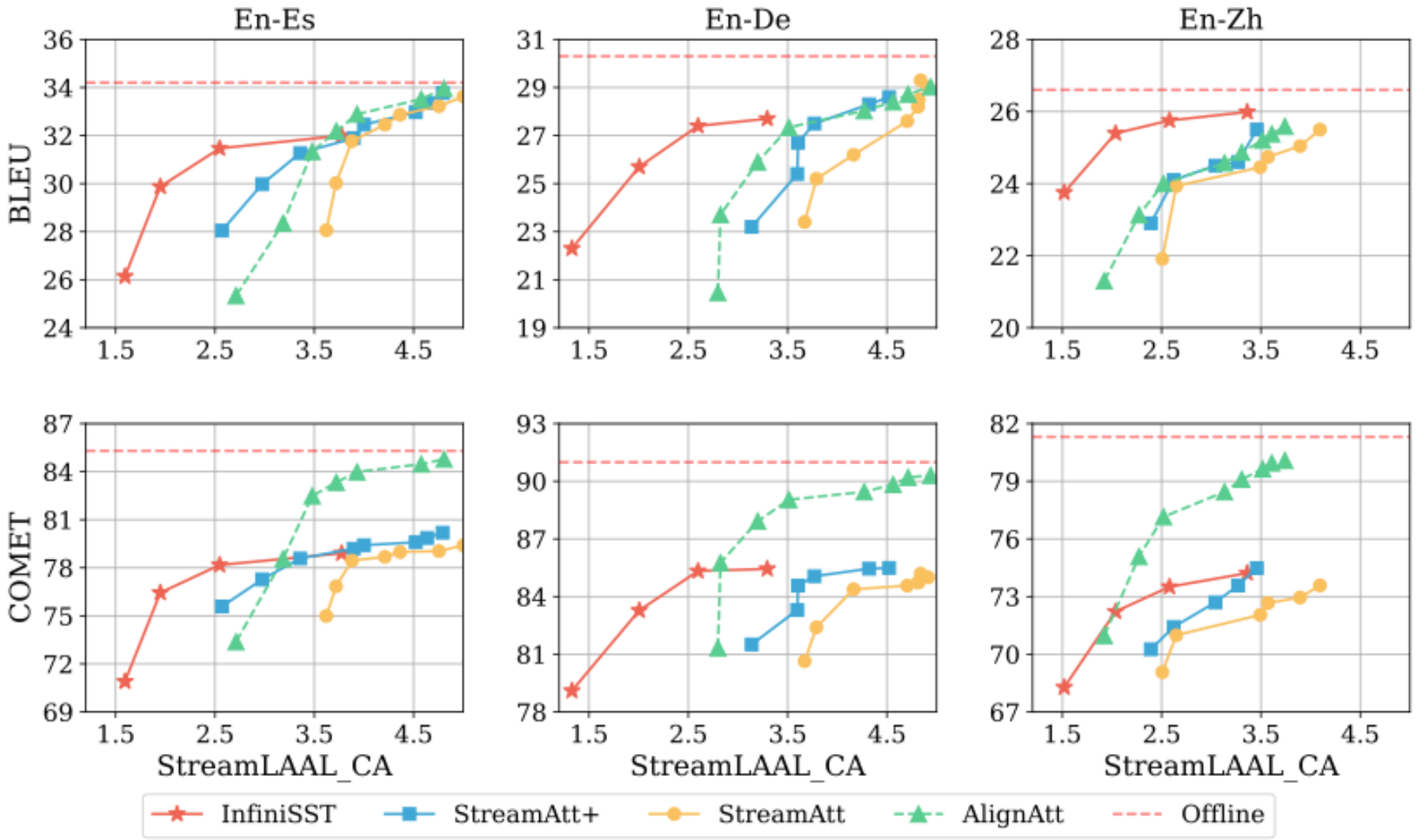
Evaluation Metrics

- Quality
  - BLEU & COMET
- Latency
  - StreamLAAL: a variant of LAAL that uses mWERSegment to segment the document translation hypothesis to align with each reference sentence, then compute LAAL on each (hyp, ref) pair

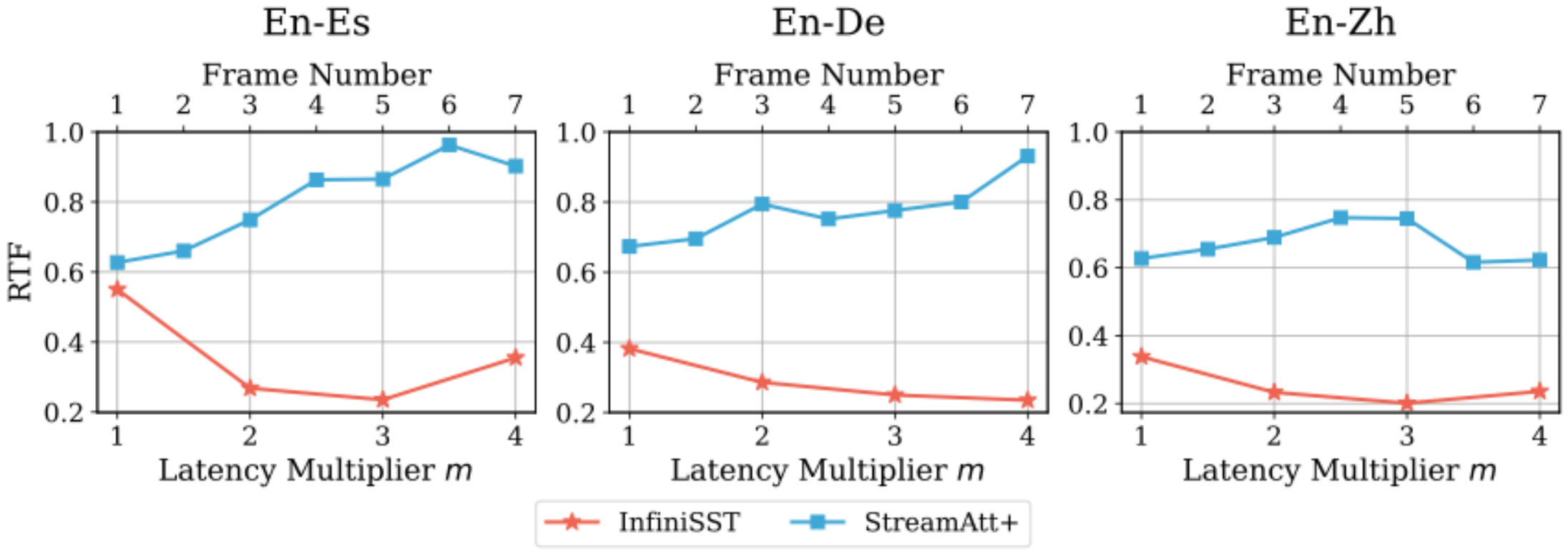
Baselines

- AlignAtt
  - Works on segmented level SST
  - Use attention scores between translation and speech to determine to stop translating or not
- StreamAtt
  - Built on top of AlignAtt, same stopping criterion
  - Preserves fixed length text history, and then cut audio history based on attention scores of preserved text
- StreamAtt+: forbid audio cutting when audio is shorter than 10 s

InfiniSST is much Faster than StreamAtt when evaluated with Computation Cost

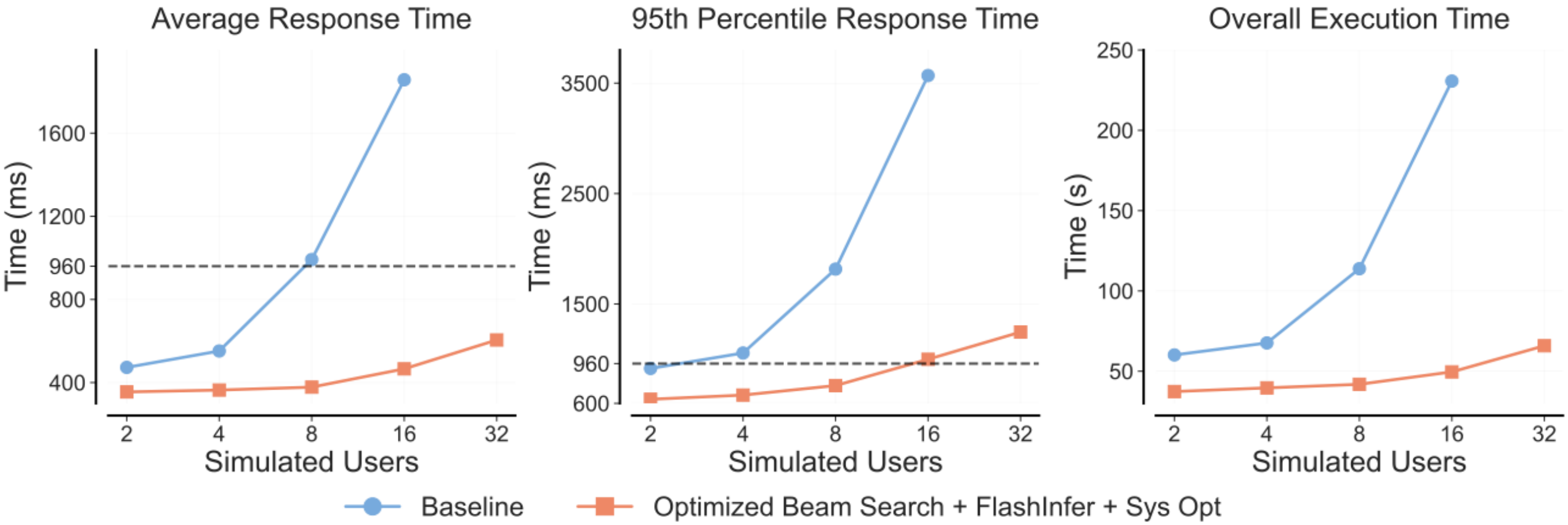


InfiniSST is much faster than StreamAtt when evaluated with Computation Cost



InfiniSST Serving System

Online Batching + Flashinfer



Highlights of InfiniSST

- Ensure translation quality:
  - pre-trained speech encoder + LLM
- Reducing latency:
  - Speech encoder: Chunk-wise unidirectional attention and in-chunk bidirectional att
  - Incremental computation ➔ avoid recomputation
  - Chat-style interleaving read/write policy
- Enable unbounded speech (long context)
  - Sliding window KV cache for speech encoder
  - System Prompt caching + sliding window cache for LLM decoder