LLMs Can Achieve High-quality Simultaneous Machine Translation as Efficiently as Offline

Biao Fu[1,3,*], **Minpeng Liao**[2,†], **Kai Fan**[2,†], **Chengxi Li**[2],
Liang Zhang[1], **Yidong Chen**[1,3], **Xiaodong Shi**[1,3,†]
[1]School of Informatics, Xiamen University [2]Tongyi Lab
[3]Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan (Xiamen University), Ministry of Culture and Tourism
biaofu@stu.xmu.edu.cn,mandel@xmu.edu.cn
{minpeng.lmp,xiji.lcx,k.fan}@alibaba-inc.com

Abstract

When the complete source sentence is provided, Large Language Models (LLMs) perform excellently in offline machine translation even with a simple prompt "*Translate the following sentence from [src lang] into [tgt lang]:*". However, in many real scenarios, the source tokens arrive in a streaming manner and simultaneous machine translation (SiMT) is required, then the **efficiency** and **performance** of decoder-only LLMs are significantly limited by their auto-regressive nature. To enable LLMs to achieve high-quality SiMT as efficiently as offline translation, we propose a novel paradigm that includes constructing supervised fine-tuning (SFT) data for SiMT, along with new training and inference strategies. To replicate the token input/output stream in SiMT, the source and target tokens are rearranged into an interleaved sequence, separated by special tokens according to varying latency requirements. This enables powerful LLMs to learn read and write operations adaptively, based on varying latency prompts, while still maintaining efficient auto-regressive decoding. Experimental results show that, even with limited SFT data, our approach achieves state-of-the-art performance across various SiMT benchmarks, and preserves the original abilities of offline translation. Moreover, our approach generalizes well to document-level SiMT setting without requiring specific fine-tuning, even beyond the offline translation model[1].

1 Introduction

Simultaneous machine translation (SiMT) (Gu et al., 2017) is a critical technique for enabling seamless cross-linguistic communication in real-time scenarios, such as international conferences. Unlike offline machine translation (OMT), where the entire source sentence is available before translation begins, SiMT systems start translating before receiving the complete input, achieving a balance between translation quality and latency.

English: Avalanche at Washington state ski resort kills 1, traps 5
Chinese: 华盛顿州滑雪胜地发生雪崩，造成 1 人死亡，5 人被困

Fixed policy (read 4 words per step by Conversational SimulMT)
Avalanche at Washington state 华盛顿州的雪崩
ski resort kills 1, 在滑雪度假胜地造成1人死亡，
traps 5 5人被困 **BLEU: 42.03**

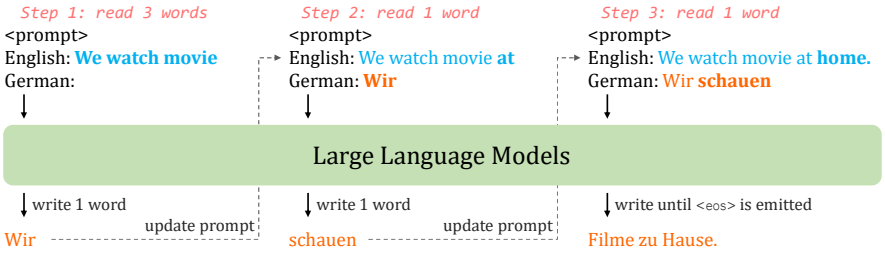
Adaptive policy (use our *EAST* method)
Avalanche at Washington state ski resort 华盛顿州滑雪场发生雪崩
kills 1, traps 5 造成1人死亡，5人被困 **BLEU: 72.04**

Figure 1: Examples of fixed policy and adaptive policy. The fixed policy (Conversational SimulMT) reads a predetermined number of words per step, leading to translation errors due to incomplete semantic understanding (e.g., prematurely translating "Avalanche at Washington state" as "华盛顿州的雪崩"). In contrast, EAST uses a adaptive policy and recognizes the incompleteness of the semantic context and continues reading more words until "ski resort", resulting in the correct translation.

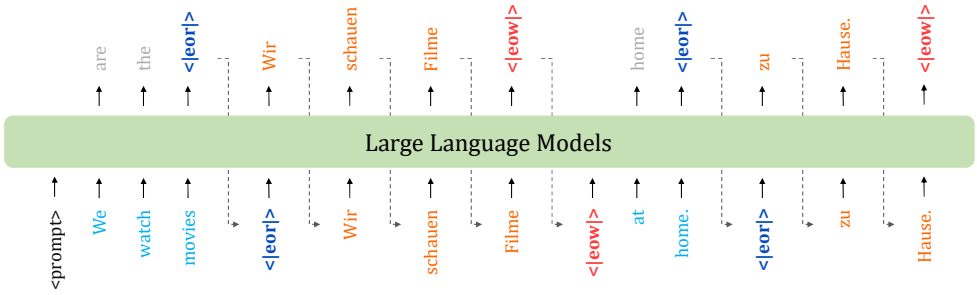
Large language models (LLMs) have achieved significant advances in OMT, demonstrating impressive capacity when translating full sentences in offline settings (Xu et al., 2024a,b; Ye et al., 2025). However, their application to SiMT remains underexplored and faces several significant challenges. First, most existing SiMT models (Zhao et al., 2023; Guo et al., 2024b; Raffel et al., 2024) are typically trained on OMT data due to the scarcity of SiMT-specific datasets. This training setup does not align well with the demands of SiMT, which hinders the model's ability to learn how to translate effectively with incomplete input (Wang et al., 2023b; Sakai et al., 2024). Second, many SiMT approaches focus on optimizing prompt structures to simulate SiMT for LLMs (Wang et al., 2023a; Koshkin et al., 2024a,b; Guo et al., 2024b; Agostinelli et al., 2024; Cheng et al., 2024), as shown in Figure 2(a), which requires recomputing the key-value (KV) cache as the prompt

[*] Work done during Biao Fu's internship at Tongyi Lab.
[†] Corresponding author.
[1]The data and code are available at https://github.com/biaofuxmu/EAST.

Findings of the Association for Computational Linguistics: ACL 2025, pages 20372–20395



(a) Previous LLM SiMT with R/W policy, *e.g.*, wait-3: $p(y_{t-2}|\mathbf{x}_{\leq t}, \mathbf{y}_{\leq t-3})$, where KV cache needs to re-calculate as t increments.



(b) The inference process of EAST in auto-regressive manner $p(\mathbf{c}_t^y|\mathbf{c}_1^x, \mathbf{c}_1^y, ..., \mathbf{c}_t^x)$, where $\mathbf{c}_t^i$ is the t-th chunk of source or target.

Figure 2: Comparison between existing LLM-based SiMT methods and our EAST. (a) Existing LLM-based SiMT typically reuses the sequence organization $\mathbf{x}_{\leq t}, \mathbf{y}_{\leq t'}$, leading the prompt to generate new target tokens always changing. (b) Our inference follows the nature of auto-regressive decoding without recalculating the KV cache. For simplicity, we use <|eor|> and <|eow|> to represent the special tokens without hurting readability.

changes continuously with the update of the source and target. This recomputation significantly increases the computational cost and inference latency, limiting the efficiency of SiMT systems (Raffel et al., 2024). Lastly, LLMs-based methods typically employs fixed policies (Wang et al., 2023a; Agostinelli et al., 2024; Sakai et al., 2024; Wang et al., 2024; Raffel et al., 2024), such as the wait-k, for their simplicity. However, these methods fail to adaptively adjust its read/write actions based on sentence structure and context, leading to suboptimal translation quality, as shown in Figure 1.

In this paper, we introduce **EAST**, an **E**fficient and **A**daptive **S**imultaneous machine **T**ranslation method with LLMs, which aims to achieve high-quality SiMT as efficiently as offline translation. Specifically, we first leverage the instruction-following capability of LLMs to generate the SiMT data (Sakai et al., 2024) with different latency levels (low, medium, and high). Two SiMT datasets are constructed for supervised fine-tuning (SFT), including the German-English dataset **SiMT-De-En-660K** and the multilingual dataset **SiMT-Multi-90K**. We structure the SFT data by alternating between the source and target segments of the generated SiMT data and introducing two special tokens (<|end-of-read|> and <|end-of-write|>) as explicit read-write signals. By performing SFT on this structured data on the LLMs, it can learn to effectively determine when to read more source input and when to generate the translation. A two-stage fine-tuning process is conducted to enhance its multilingual translation capabilities, with full-weight fine-tuning on the SiMT-De-En-660K dataset, followed by LoRA (Hu et al., 2022) fine-tuning on a combination of the SiMT-Multi-90K and Off-Multi-120K datasets. During inference, EAST employs an adaptive read-write policy that aligns with its SFT recipe. The model predicts token-by-token and then dynamically switches between read and write actions based on whether the predicted token is a read or write signal. Due to the auto-regressiveness of our token input/output (I/O) sequence, where new source input and target translations are incrementally appended, EAST can efficiently reuse the KV cache without modifying historical sequence. This significantly reduces computational costs and inference latency, improving the overall efficiency of SiMT. A comparison to highlight the main difference between the previous LLM SiMT and ours is shown in Figure 2. Experimental results show that EAST achieves high-quality SiMT across eight translation directions and near-offline decoding speeds, without compromising offline translation performance, and generalizes well to document-level SiMT.

Our contributions are summarized as follows:

- We construct two novel latency-aware datasets, including a German-English dataset (SiMT-De-En-660K) and a multilingual dataset (SiMT-Multi-90K), where latency-awareness is often neglected in SiMT studies.
- We propose a novel LLM-based adaptive read/write policy which achieves high-quality SiMT as efficiently as offline model. To the best of our knowledge, this is the first efficient and adaptive LLM-based SiMT method.
- Experimental results on multilingual and document-level test sets demonstrate the effectiveness of our method, where document-level evaluation is particularly underexplored in prior work.
- Our findings reveal that only 10K SiMT examples may be sufficient to achieve commendable translation quality, offering valuable insights for future research.

2 Related Work

Traditional SiMT SiMT requires starting translation before the full source sentence is available, aiming to balance translation quality and latency. To achieve this, a read-write policy is introduced to determine whether to wait for more input or begin translating. Traditional SiMT models are often built on encoder-decoder architectures, with fixed or trainable policies. The widely studied fixed wait-k policy (Ma et al., 2019; Elbayad et al., 2020; Zhang and Feng, 2021; Zhang et al., 2023a; Fu et al., 2023) is simple but struggles with complex contexts or non-monotonic language pairs (Zhang et al., 2022). Adaptive policies, which dynamically determine when to read and write, offer improved translation quality. To enable models to learn effective read-write decisions, a variety of techniques have been applied, including reinforcement learning (Gu et al., 2017; Arthur et al., 2021; Miao et al., 2021), dynamic programming (Miao et al., 2021; Liu et al., 2021; Fu et al., 2024), data augmentation (Zhang et al., 2020; Deng et al., 2023), information transport theory (Zhang and Feng, 2022), Hidden Markov Models (Zhang and Feng, 2023), and decoder-only architecture (Guo et al., 2024a).

LLM-based SiMT Recent studies have explored leveraging LLMs for SiMT, but traditional adaptive policies in encoder-decoder architectures not well-suited for LLMs. Some approaches (Wang et al., 2023a; Koshkin et al., 2024a,b; Agostinelli et al., 2024) optimize prompts with a fixed wait-k policy. Others, like the Agent-SiMT (Guo et al., 2024b), combine traditional adaptive SiMT models to guide read/write decisions. However, these approaches face issues: updating prompts during inference prevents KV cache reuse, leading to increased recomputation and latency, and Agent-SiMT requires training an additional SiMT model, complicating the process. Recent efforts (Sakai et al., 2024; Cheng et al., 2024) use LLMs to generate SiMT data for adaptive policy learning but still struggle with inefficiencies in LLM-based SiMT systems.

To improve translation efficiency, Wang et al. (2024) introduce the Conversational SimulMT framework, which employs a multi-turn dialogue decoding approach with generating SFT data by segmenting parallel sentences with an alignment tool. However, the method employs a fixed policy during inference that reads a fixed number of words at each step, leading to a mismatch with the fine-tuning process. The differences between Conversational SimulMT and our method are described in the Appendix A. Moreover, SimulMask (Raffel et al., 2024) improves prompt-based efficiency by introducing a policy-specific attention mask during fine-tuning. It mimics inference behavior, limiting target token attention to the relevant source prompt. However, its complex masking requires prior knowledge of policy decisions, making it unsuitable for adaptive policies.

In addition, these SiMT methods primarily focus on leveraging the translation capabilities of LLMs, without exploring the adaptive read-write policies and generalization abilities. Unlike previous methods, EAST enables the LLMs to learn adaptive read-write policies in various latency requirements, and utilizes an interleaved text structure to significantly improve the inference efficiency of the LLMs while maintaining consistency between fine-tuning and inference.

3 Methods

In this paper, we propose EAST, an Efficient and Adaptive Simultaneous machine Translation method with LLMs, which involves three key components: the construction of SiMT data, the training of the LLMs on the SFT data, and the inference with adaptive read-write policy.

3.1 SiMT Data Curation via Latency-aware Chunk Segmentation

The availability of SiMT-specific datasets is scarce, and the annotation by professional interpreters is time-consuming and expensive. To address this issue, we leverage the powerful instruction-following capability of LLMs after RLHF (Ouyang et al., 2022) (*e.g.*, GPT-4 (OpenAI et al., 2024)) and design a prompt that instructs LLMs to act as a professional simultaneous interpreter, segmenting sentences into independent semantic chunks and generating corresponding translations for each chunk.

In practice, SiMT must accommodate varying latency requirements depending on different use cases, such as live broadcasts that prioritize low latency and formal conferences that demand high-quality translation with higher latency. Importantly, different latencies naturally influence how sentences are segmented, reordered, and translated. Therefore, we prompt LLMs to generate SiMT data at three latency levels: "low", "medium", and "high". The prompt template is provided in Figure 19 in Appendix. Concretely, given a language pair $\mathbf{x}_{1:T_x}, \mathbf{y}_{1:T_y}$, the LLM output of the proposed low latency prompt can be represented as follows:

$$\mathbf{x}_{1:T_x} = [\mathbf{c}_1^x, \cdots, \mathbf{c}_{T_{low}}^x], \quad (1)$$

$$\mathbf{y}_{1:T_y} = [\mathbf{c}_1^y, \cdots, \mathbf{c}_{T_{low}}^y], \quad (2)$$

where $\mathbf{c}_t^{[\cdot]}$ is the t-th semantic chunk of source or target. With simple length filtering, the two chunk sequences should be well aligned with the same length. Similarly, we can obtain the medium and high latency output. In general, for the same pair, we have $T_{low} \geq T_{medium} \geq T_{high}$.

In this study, we curated a dataset of 660K SiMT samples by extracting language pairs from the WMT15 De-En training data, allocating one-third of the samples to each latency requirement. While existing LLM-based SiMT methods typically train separate models for different language pairs (Guo et al., 2024b; Raffel et al., 2024), they often overlook the inherent multilingual capabilities of LLMs. In contrast, we constructed a smaller multilingual SiMT dataset of 90K samples encompassing eight translation directions.

3.2 Training LLMs with SFT

To tackle the challenge proposed at the beginning of this section, we propose a two-stage SFT training process on the two curated SiMT datasets.

Stage I: Activate SiMT of LLMs The objective of this stage is to teach the LLMs how to perform adaptive simultaneous translation by learning when to read and write in our designed format. To enable the model to learn these adaptive behaviors, we reorganized the aligned chunks in the SiMT data by interleaving between source and target chunks and introduce two special tokens (<|end-of-read|> and <|end-of-write|>), *i.e.*,

$$[\mathbf{c}_1^x, \texttt{<|eor|>}, \mathbf{c}_1^y, \texttt{<|eow|>}, \cdots, \mathbf{c}_T^x, \texttt{<|eor|>}, \mathbf{c}_T^y, \texttt{<|eow|>}]. \quad (3)$$

The special tokens act as explicit signals for the model to transition between reading and writing. The SiMT annotation process shows that each chunk contains enough semantic meaning for LLMs to carry out translations, ensuring that sequence reorganization does not lead to any loss of information for the model's reading or writing decisions. Since the annotation process also encodes the degree of fragmentation into latency indicator tokens—"low", "medium" or "high", the SFT can effectively guide the model to adapt to varying latency requirements. Figure 18 provides a comprehensive example of the SFT data.

We train for one epoch during this stage on the larger SiMT-De-En-660K, employing full parameter tuning. As SiMT defined in our proposed format is generally a novel task for LLMs, full parameter tuning ensures that the LLM can effectively and successfully learn the auto-regressive SiMT. Training for just one epoch helps mitigate the risk of overfitting during the full parameter tuning.

Stage II: Generalize to Multilingual SiMT As the LLM acquires its auto-regressive SiMT capability in Stage I, its inherent multilingual proficiency enables it to generalize to multilingual SiMT, even with limited SFT data. Consequently, we apply LoRA (Hu et al., 2022) fine-tuning to a smaller multilingual dataset of 90K instances including eight language directions. Additionally, during this stage, we incorporate an OMT task to bolster the model's ability to translate full sentences and enhance overall translation performance. In fact, we can view offline translation as a specific instance of SiMT by treating the entire sentence as a complete semantic chunk, *e.g.*, $\mathbf{x}_{1:T} = \mathbf{c}_1^x$.

Loss In previous LLM-based SiMT methods (Wang et al., 2024), loss calculation *w.r.t.* source text is typically masked out, as it does not contribute to the training. However, in our case, the

cross-entropy loss is calculated on both the target text and the source text, as well as on special tokens. The primary goal is to align with the auto-regressive design of interleaved sequences and establish the appropriate reading and writing timing.

3.3 Efficient Inference for Adaptive SiMT

Unlike Conversational SimulMT (Wang et al., 2024), which adopts a fixed policy (e.g., reading a fixed number of tokens at each step) during inference and suffers from a mismatch between training and inference phases, EAST performs auto-regressive token-by-token prediction aligned with the training process as shown in Figure 2(b). The process unfolds in two main phases:

Read-Predict-Discard During the read phase, the model sequentially receives source tokens and predicts the next token. If the predicted token is not <|end-of-read|>, it is discarded, and the next source token is appended to the current source chunk. Once <|end-of-read|> is predicted, the model transitions to the translating phase. Note that the discarding operation in the read phase does not violate the incremental appending of contexts, enabling EAST to efficiently utilize KV-cache for faster generation.

Predict-Append Once the model enters the translation phase, it directly begins predicting the next token. If the predicted token is not <|end-of-write|>, it is appended to the current target chunk. When <|end-of-write|> is predicted, the model completes the current translation and returns to the reading phase.

Similar to the training phase, inference controls latency through indicator tokens—"low", "medium", or "high". Interestingly, an **Interpolation Effect** is observed, allowing for generalization to other latency levels using indicator tokens such as "low-medium" or "medium-high". Consequently, we can gather 3 to 5 observations to draw the BLEU-AL curve.

4 Experiments

4.1 Experimental Settings

Datasets Following ALMA (Xu et al., 2024a), we collect test data from WMT17 to WMT21, covering the 8 language directions: De↔En, Zh↔En, Ru↔En, and Cs↔En, and refer to this collection as **Off-Multi-120K** for OMT training. As introduced in the previous section, the primary SiMT SFT dataset to initiate novel task learning is **SiMT-De-En-660K**, derived from the WMT15 De→En training dataset. In addition, we construct a smaller multilingual SiMT SFT dataset, **SiMT-Multi-90K**, derived from Off-Multi-120K dataset[2]. As shown in Table 5 of Appendix, the **sentence-level test data** is extracted from WMT22 across the same 8 translation directions as the OMT data. The majority of existing research primarily focuses on sentence-level evaluation. However, in many real-world applications, such as speech delivery, the input for SiMT often comes at the document level rather than isolated sentences. Moreover, LLMs have demonstrated impressive capabilities in long-form generation. Thus, we directly evaluate EAST on WMT22 **document-level test data** without additional fine-tuning.

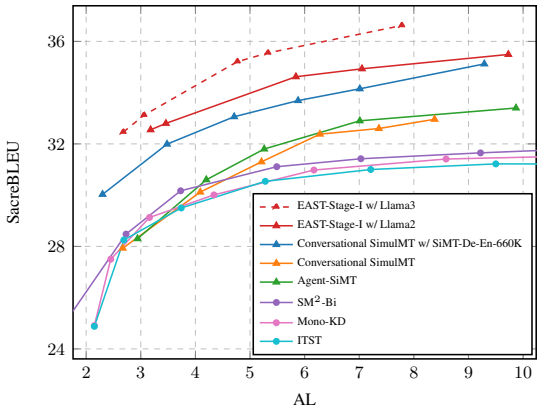


Figure 3: BLEU vs. AL on WMT15 De→En test set.

Metrics For quality evaluation, we use automatic metric–SacreBLEU[3] to compute the corpus-level BLEU, along with neural evaluation metrics BLEURT[4] (Sellam et al., 2020; Pu et al., 2021) and COMET[5] (Rei et al., 2020, 2022). For latency evaluation, we adopt Average Latency (AL) (Ma et al., 2019), computation-aware AL (AL-CA)[6], and Length-Adaptive AL (LAAL) (Papi et al., 2022). In addition, we use Word Wall Time (WWT) (Wang et al., 2024) to evaluate the model's decoding speed by calculating the actual inference time per word. For the implementation details of our model, please refer to Appendix D.

System Settings In this paper, we conduct comparative experiments between EAST and the following baselines. We employ the same training setup as

[2]See Appendix C for details of data processing.

[3]https://github.com/mjpost/sacrebleu

[4]BLEURT-20

[5]wmt22-comet-da

[6]The AL-CA metric is calculated by adding the machine processing time to the policy delay .

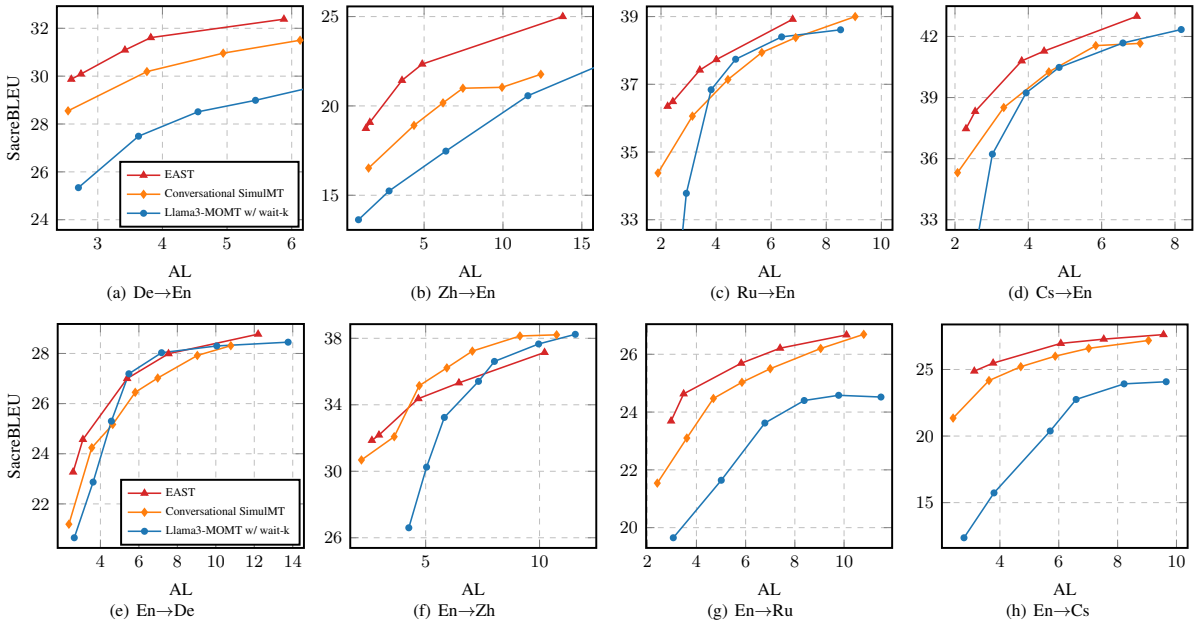


Figure 4: SacreBLEU against AL on the WMT22 X→En and En→X test sets.

EAST in these baselines.

- **EAST**: The proposed pipeline includes two-stage training, *i.e.*, full-weight fine-tuning on SiMT-De-En-660K followed by LoRA fine-tuning on SiMT-Multi-90K and Off-Multi-120K datasets.
- **EAST-Stage-I**: Full-weight fine-tuning on the SiMT-De-En-660K dataset.
- **Conversational SiMT** (Wang et al., 2024): It first generates SiMT data by segmenting parallel sentences using an alignment tool, and then formats it into a multi-round dialogue prompts for SFT. During inference, it reads k tokens each step and then incrementally decodes them.
- **Llama3-MOMT**: LoRA fine-tuning on the Off-Multi-120K dataset for OMT using Llama-3-8B-Instruct as the base model.
- **Llama3-MOMT w/ wait-k**: Applying the wait-k policy on the trained **Llama3-MOMT** model for streaming inference.

4.2 Main Results

SiMT WMT15 De→En To compare with the previous SOTA models, We first evaluate our method on the WMT15 De→En test set. As Figure 3 shows, we compare our method with two categories of baselines: (1) Traditional SiMT methods, including ITST (Zhang and Feng, 2022), Mono-KD (Wang et al., 2023b), and SM^2-Bi (Yu et al., 2024); (2) LLM-based SiMT methods, including Agent-SiMT (Guo et al., 2024b) and Conversational SimulMT (Wang et al., 2024). Our EAST-Stage-I achieves superior BLEU-AL curves, outperforming these traditional approaches with a large margin. We admit that the LLMs are typically pre-trained on extensive multilingual corpora, giving them an inherent advantage over smaller SiMT models. However, it is important to recognize that our definition of auto-regressive SiMT with an adaptive policy represents a completely novel challenge for LLMs, and the size of our SFT dataset considerably smaller than that of these methods. In addition, even the recent LLM-based SiMT method Conversational SiMT and Agent-SiMT can only achieve on-par performance with traditional SiMT methods and does not show significant advantages.

When Conversational SiMT is trained on our SiMT-De-En-660K dataset, it achieves more than a 2 BLEU improvement across all latency settings compared to its original counterpart with a larger dataset (4M examples). This demonstrates the effectiveness of our dataset. When incorporating our adaptive policy (EAST-Stage-I), we observe an additional 0.9 BLEU improvement, showing the effectiveness of our policy. Upgrading the backbone model from Llama2 to Llama3 results in +0.5 BLEU in low-latency regions and +1 BLEU in high-latency regions.

Method	Low	Low-Medium	Medium	Medium-High	High
Conversational-SiMT	47.85 / 2.18	47.29 / 3.62	45.68 / 4.72	43.04 / 5.93	32.70 / 10.75
EAST	38.98 / 2.64	38.60 / 2.95	35.25 / 4.68	33.88 / 6.46	32.12 / 10.22

Table 1: PPL and AL results on the WMT22 En→Zh test set at different latency settings.

Models	De→En		Zh→En		Ru→En		Cs→En		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
GPT-4	33.87	85.62	27.20	82.79	43.51	86.18	48.67	87.43	38.31	85.51
Bayling-7B	28.16	83.19	20.31	77.48	34.74	82.48	35.98	82.03	29.80	81.30
ALMA-7B-LoRA	29.56	83.95	23.64	79.78	39.21	84.84	43.49	85.93	33.98	83.63
Llama3-MOMT	31.98	**84.89**	**25.48**	**81.26**	**39.83**	**85.19**	44.92	**86.23**	**35.55**	**84.39**
EAST	**32.55**	84.77	23.80	80.86	**39.83**	85.04	**45.61**	86.20	35.45	84.22

Models	En→De		En→Zh		En→Ru		En→Cs		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
GPT-4	35.38	87.44	43.98	87.49	30.45	88.87	34.53	90.77	36.09	88.64
Bayling-7B	25.66	82.18	38.19	84.43	14.85	74.72	15.64	76.85	23.59	79.55
ALMA-7B-LoRA	30.16	85.45	36.47	84.87	**26.93**	87.05	**30.17**	**89.05**	30.93	86.61
Llama3-MOMT	30.45	**85.63**	**40.68**	**86.53**	24.83	**87.27**	27.92	88.36	30.97	**86.95**
EAST	**30.84**	85.49	40.17	86.31	26.79	87.13	26.63	88.17	**31.11**	86.78

Table 2: Offline results on the WMT22 X→En and En→X test sets.

SiMT WMT22 X↔En We further evaluate EAST on the WMT22 X↔En test sets, as shown in Figure 4. We compare EAST with two LLM-based baselines: Llama-MOMT w/ wait-k and Conversation SimulMT, where Conversation SimulMT is reproduced based on the Llama3 backbone model with EAST's SFT data and the two-stage training method for fair comparisons. Llama-MOMT w/ wait-k shows inferior translation quality in all latency areas due to the limitations of the fixed policy. Compared to Conversation SimulMT, EAST achieves higher BLEU in all latency regions across eight directions, with an average improvement of 1.5 BLEU in low latency regions. In the En→Zh direction, while Conversational SimulMT achieves a higher BLEU at high latency area, it is lower than EAST in COMET and BLEURT metrics, as shown in Figures 8 and 9. These metrics better reflect semantic quality that better correlate with human judgments, whereas BLEU primarily measures surface-level n-gram overlap with the reference. This result can be attributed to the fixed policy used in Conversational SimulMT. It often generates translations before the complete semantic context is available, which can lead to surface-level matches with the reference (thus increasing BLEU), but at the cost of semantic completeness or fluency—resulting in lower COMET and BLEURT scores. To further verify this, we use `Llama-3-8B` to compute the PPL of translations. As shown in Table 1, EAST achieves lower PPL scores across all latency levels, demonstrating that its translations are more fluent and semantically coherent. A detailed comparison of the COMET and BLEURT metrics is provided in the Appendix E.1.

Offline Performance We also evaluate the performance of offline translation on the WMT22 test set, as presented in Table 2. Our results are superior to previous studies, Bayling (Zhang et al., 2023b) and ALMA (Xu et al., 2024a), except for a slight lag in En→Cs. Compared to OMT model Llama3-MOMT and other variants, EAST maintains comparable or superior offline translation performance across the eight language directions, indicating that our two-stage SFT process effectively maintains translation quality for OMT.

In summary, these results highlight that EAST not only excels in high-quality simultaneous translation but also ensures that the offline translation capabilities are not compromised.

4.3 Zero-Shot Generalization to Document-level SiMT

Since real-world applications often involve streaming inputs that are typically long and unsegmented, we further evaluate the EAST directly on the document-level test set from WMT22 De/Ru→En without fine-tuning on document-level data. In our

Method	BLEU (↑)	AL (↓)	AL-CA (↓)	WWT (ms) (↓)
EAST-offline	32.55	14.62	15.22	38.96
EAST	29.87/31.08/32.38	2.59/3.42/5.87	3.26/4.29/6.55	49.87 (±1,21)
Llama3-MOMT w/ wait-k	26.50/27.60/28.95	2.70/3.63/5.44	3.69/4.69/6.43	977.2 (±4.49)

Table 3: Comparison of inference latency and speed on WMT22 De→En test set. The BLEU, AL, and AL-CA scores are given for low, medium, and high latency settings respectively. WWT refers to the actual inference time per word and is reported as mean and standard deviations (in parentheses) over the three latency.

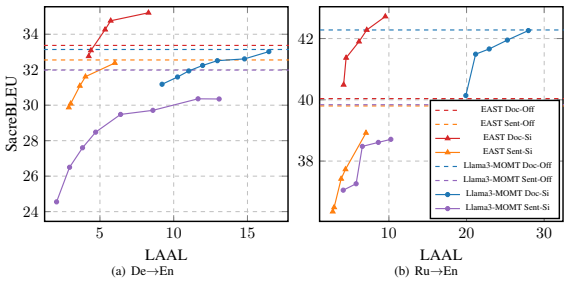


Figure 5: SacreBLEU-LAAL curves on the WMT22 document-level De/Ru→En test set. Methods labeled with "-Off" refer to offline translation, *i.e.*, including the entire document in the prompt. Methods marked with "-Si" denote simultaneous translation, involving the streaming input.

experiments, the document-level test set is derived from the same data as the sentence-level set but without sentence segmentation. The results *w.r.t.* the corpus BLEU are depicted in Figure 5. EAST shows superior performance in document-level settings, as this enhancement is due to the model's improved ability to leverage historical context, thus enhancing translation accuracy and coherence.

Originally, document-level offline translation was expected to be one of the strongest capabilities of LLMs. Surprisingly, our proposed EAST model significantly outperforms both EAST and Llama3-MOMT in offline performance within a document-level context. This discrepancy in previous works may arise from models being trained exclusively on sentence-level data, which can lead to a training-inference mismatch during offline translation. Additionally, the long context of the source document may contribute to forgetting issues during the generation of the target document. However, our training approach, which alternates between source and target texts, effectively minimizes these mismatches. These results indicate that EAST is better suited for longer text sequences, making it particularly suitable for streaming scenarios.

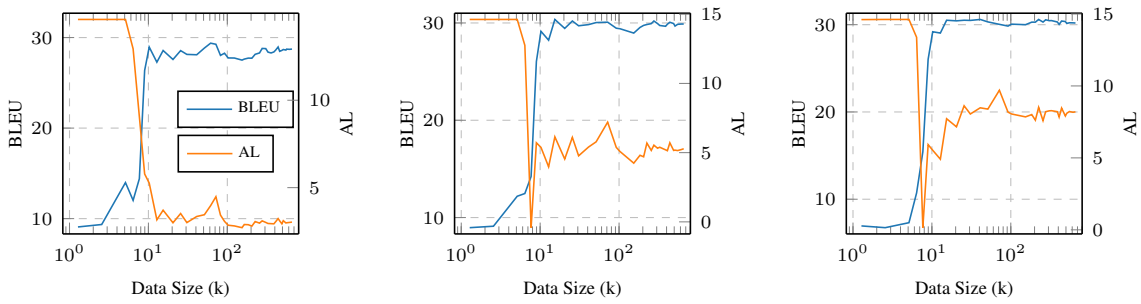
Moreover, it can be observed that there is a significant rightward shift on the document-level BLEU-LAAL curve of the wait-k method (Llama3-MOMT Doc-Si) compared to its sentence-level counterpart (Llama3-MOMT Sent-Si). Our statistical data indicates that English texts are substantially longer than their German and Russian counterparts in document-level test set—averaging 16.1 words longer than German and 35.8 words longer than Russian. This discrepancy is much greater than in the sentence-level test set, where English texts are only 2.2 and 3.1 words longer, respectively. Instead, EAST uses an adaptive read/write policy that effectively mitigates the above problems.

4.4 Inference as Efficient as Offline

In this section, we measure the overall efficiency of the EAST model on an NVIDIA A100 through computation-aware latency (AL-CA) and decoding speed (WWT), as shown in Table 3. EAST achieves comparable translation performance to its offline counterpart EAST-offline with lower latency, and significantly outperforms Llama3-MOMT w/ wait-k method under similar latency conditions. For decoding speed, Llama3-MOMT w/ wait-k shows the slowest inference speeds, taking up to 977.2ms to generate a single word. This inefficiency is attributed to the inability of this method to efficiently utilize the KV cache, necessitating the re-encoding of historical content at each decoding step, which limits its practical use in real-time scenarios. Conversely, EAST efficiently leverages KV-cache during inference, taking only 49ms to decode a word, achieving comparable decoding speeds to its offline counterpart (38.96ms per word). This small difference of about 10ms shows that EAST maintains near-offline efficiency, even under the streaming input conditions of SiMT.

4.5 How Many Examples Are Needed to Teach LLMs the Novel SiMT Task?

In this section, we investigate the data size required for efficiently training LLMs on the novel SiMT task based on the SiMT-De-En-660K dataset. Figure 6 illustrates the changes in BLEU and AL



(a) Low Latency (b) Medium Latency (c) High Latency

Figure 6: BLEU scores (left y-axis) and AL values (right y-axis) over data size. We use log scale for scale the x-axis to more clearly observe the effect of data size. The original plots are also provided in Figure 7.

scores with data sizes for different latency settings– "low", "medium", and "high". First, there is a significant improvement in BLEU score as the data size increases to about 10K. Beyond this point, the rate of increase in BLEU score diminishes. Similarly, AL metrics also decrease notably as the data size reaches around 10k, before stabilizing or showing minor fluctuations. This pattern is consistent across all latency settings, indicating a general learning behavior of the model: rapid enhancements in translation quality are achieved with the first 10K examples, followed by a phase where the model focuses on refining its read/write policy across different latency levels, up to 100k examples.

The results suggest that a relatively small dataset of just 10K SiMT examples may be sufficient to achieve commendable translation quality. This finding aligns well with our multilingual dataset, which contains approximately 10K examples per language, facilitating good performance across different languages. Moreover, expanding the data size (up to 100K examples) can further optimize the model's read/write policy.

More analysis are included in the Appendix. Appendix E.2 analyzes the performance of EAST under various training strategies. Appendix E.3 compares the hallucination rate of different methods. Appendix E.4 evaluates the quality of different translation policies. Appendix E.5 explores the impact of different backbones on translation performance. Appendix E.6 compares the SiMT performance of EAST with GPT-4. Appendix E.7 evaluates the fluency of EAST translations.

5 Conclusion

In this paper, we introduce an **E**fficient and **A**daptive **S**imultaneous **T**ranslation method using LLMs, EAST, designed to achieve high-quality SiMT with the efficiency of offline systems. By constructing SFT data, leveraging an interleaved token structure with explicit read-write signals and incorporating latency-aware prompts, EAST enables LLMs to perform adaptive reading and translation based on varying latency requirements. Our experimental results demonstrate that EAST not only achieves state-of-the-art performance on SiMT benchmarks but also maintains high-quality translations in offline settings. Additionally, EAST shows excellent generalization to document-level SiMT, highlighting its suitability for streaming translation in real-world scenarios.

Limitations

The proposed method assumes an idealized setting where the input is clean and fluent. In real-world applications, however, simultaneous translation often involves noisy or disfluent input. The model's performance under such conditions has not been evaluated. Additionally, our approach is designed and evaluated for simultaneous text-to-text translation tasks, leaving the domain of simultaneous speech-to-text translation unexplored.

Acknowledgements

We would like to thank all the anonymous reviewers for the insightful and helpful comments. This work is supported by the National Science and Technology Major Project (Grant No. 2022ZD0116101), the Major Scientific Research Project of the State Language Commission in the 13th Five-Year Plan (Grant No. WT135-38), the public technology service platform project of Xiamen City (No. 3502Z20231043), and Alibaba Research Intern Program.

References

Victor Agostinelli, Max Wild, Matthew Raffel, Kazi Fuad, and Lizhong Chen. 2024. Simul-LLM: A framework for exploring high-quality simultaneous translation with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10530–10541, Bangkok, Thailand. Association for Computational Linguistics.

Philip Arthur, Trevor Cohn, and Gholamreza Haffari. 2021. Learning coupled policies for simultaneous machine translation using imitation learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2709–2719, Online. Association for Computational Linguistics.

Junkun Chen, Renjie Zheng, Atsuhito Kita, Mingbo Ma, and Liang Huang. 2021. Improving simultaneous translation by incorporating pseudo-references with fewer reorderings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5857–5864, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shanbo Cheng, Zhichao Huang, Tom Ko, Hang Li, Ningxin Peng, Lu Xu, and Qini Zhang. 2024. Towards achieving human parity on end-to-end simultaneous speech translation via llm agent. *arXiv preprint arXiv:2407.21646*.

Hexuan Deng, Liang Ding, Xuebo Liu, Meishan Zhang, Dacheng Tao, and Min Zhang. 2023. Improving simultaneous machine translation with monolingual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12728–12736.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2020. Efficient wait-k models for simultaneous machine translation. *arXiv preprint arXiv:2005.08595*.

Biao Fu, Kai Fan, Minpeng Liao, Yidong Chen, Xiaodong Shi, and Zhongqiang Huang. 2024. wav2vec-S: Adapting pre-trained speech models for streaming. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 11465–11480, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Biao Fu, Minpeng Liao, Kai Fan, Zhongqiang Huang, Boxing Chen, Yidong Chen, and Xiaodong Shi. 2023. Adapting offline speech translation models for streaming with future-aware distillation and inference. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16600–16619, Singapore. Association for Computational Linguistics.

Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. 2017. Learning to translate in real-time with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1053–1062, Valencia, Spain. Association for Computational Linguistics.

Shoutao Guo, Shaolei Zhang, and Yang Feng. 2024a. Decoder-only streaming transformer for simultaneous translation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8851–8864, Bangkok, Thailand. Association for Computational Linguistics.

Shoutao Guo, Shaolei Zhang, Zhengrui Ma, Min Zhang, and Yang Feng. 2024b. Agent-simt: Agent-assisted simultaneous machine translation with large language models. *arXiv preprint arXiv:2406.06910*.

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*.

Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024a. Llms are zero-shot context-aware simultaneous translators. *arXiv preprint arXiv:2406.13476*.

Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024b. Transllama: Llm-based simultaneous translation system. *arXiv preprint arXiv:2402.04636*.

Dan Liu, Mengge Du, Xiaoxi Li, Ya Li, and Enhong Chen. 2021. Cross attention augmented transducer networks for simultaneous translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 39–55, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. 2019. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3025–3036, Florence, Italy. Association for Computational Linguistics.

Yishu Miao, Phil Blunsom, and Lucia Specia. 2021. A generative framework for simultaneous machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6697–6706, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin,

Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Sara Papi, Marco Gaido, Matteo Negri, and Marco Turchi. 2022. Over-generation cannot be rewarded: Length-adaptive average lagging for simultaneous speech translation. In *Proceedings of the Third Workshop on Automatic Simultaneous Translation*, pages 12–17, Online. Association for Computational Linguistics.

Amy Pu, Hyung Won Chung, Ankur Parikh, Sebastian Gehrmann, and Thibault Sellam. 2021. Learning compact metrics for MT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 751–762, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Matthew Raffel, Victor Agostinelli, and Lizhong Chen. 2024. Simultaneous masking, not prompting optimization: A paradigm shift in fine-tuning llms for simultaneous translation. *arXiv preprint arXiv:2405.10443*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T.

Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Yusuke Sakai, Mana Makinae, Hidetaka Kamigaito, and Taro Watanabe. 2024. Simultaneous interpretation corpus construction by large language models in distant language pair. *arXiv preprint arXiv:2404.12299*.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Minghan Wang, Thuy-Trang Vu, Ehsan Shareghi, and Gholamreza Haffari. 2024. Conversational simulmt: Efficient simultaneous translation with large language models. *arXiv preprint arXiv:2402.10552*.

Minghan Wang, Jinming Zhao, Thuy-Trang Vu, Fatemeh Shiri, Ehsan Shareghi, and Gholamreza Haffari. 2023a. Simultaneous machine translation with large language models. *arXiv preprint arXiv:2309.06706*.

Shushu Wang, Jing Wu, Kai Fan, Wei Luo, Jun Xiao, and Zhongqiang Huang. 2023b. Better simultaneous translation with monotonic knowledge distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2334–2349, Toronto, Canada. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024a. A paradigm shift in machine translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024b. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*.

Yongshi Ye, Biao Fu, Chongxuan Huang, Yidong Chen, and Xiaodong Shi. 2025. How well do large reasoning models translate? a comprehensive evaluation for multi-domain machine translation. *Preprint*, arXiv:2505.19987.

Donglei Yu, Xiaomian Kang, Yuchen Liu, Yu Zhou, and Chengqing Zong. 2024. Self-modifying state modeling for simultaneous machine translation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9781–9795, Bangkok, Thailand. Association for Computational Linguistics.

Linlin Zhang, Kai Fan, Jiajun Bu, and Zhongqiang Huang. 2023a. Training simultaneous speech translation with robust and random wait-k-tokens strategy. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7814–7831, Singapore. Association for Computational Linguistics.

Ruiqing Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2022. Learning adaptive segmentation policy for end-to-end simultaneous translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7862–7874, Dublin, Ireland. Association for Computational Linguistics.

Ruiqing Zhang, Chuanqiang Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2020. Learning adaptive segmentation policy for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2280–2289, Online. Association for Computational Linguistics.

Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, et al. 2023b. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models. *arXiv preprint arXiv:2306.10968*.

Shaolei Zhang and Yang Feng. 2021. Universal simultaneous machine translation with mixture-of-experts wait-k policy. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7306–7317, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shaolei Zhang and Yang Feng. 2022. Information-transport-based policy for simultaneous translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 992–1013, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shaolei Zhang and Yang Feng. 2023. Hidden markov transformer for simultaneous machine translation. In *The Eleventh International Conference on Learning Representations*.

Libo Zhao, Kai Fan, Wei Luo, Wu Jing, Shushu Wang, Ziqian Zeng, and Zhongqiang Huang. 2023. Adaptive policy with wait-k model for simultaneous translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4832, Singapore. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, YeYanhan YeYanhan, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*

(Volume 3: System Demonstrations), pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

A Key Differences from Conversational SimulMT

A.1 Data Construction

Conversational SimulMT relies on an alignment tool and multi-step data augmentation to create SiMT data. However, this approach has significant limitations: **(i)** The resulting subsequences may not represent meaningful semantic units. **(ii)** The data construction method is based on offline parallel data, which introduces a domain mismatch with the SiMT paradigm. **(iii)** The word alignments generated by the alignment tool fast_align used in Conversational SimulMT have an error rate of approximately 30%, potentially degrading the quality of synthetic data and leading to suboptimal performance.

In contrast, EAST leverages GPT-4 to segment source text into independent semantic units and generate corresponding simultaneous translations, effectively mitigating these issues. Moreover, our data construction considers varying latency levels, as the models should generate different translations for different latency requirements, an important consideration often overlooked in prior works. As shown in Figure 3, when training our SiMT-De-En-660K dataset in Conversational SimulMT format, we observe a consistent performance improvement of 2 BLEU across all latency settings. Notably, our dataset (660K samples) is an order of magnitude smaller than that of Conversational SimulMT (4M samples). Moreover, the results in Figure 6 show that only 10K SiMT examples may be sufficient to achieve commendable translation quality.

This concludes that a limited amount of high-quality, latency-aware aligned data is more effective than large-scale, tool-aligned data.

A.2 Training for Adaptive Read/Write Policy

In Conversational SimulMT, the SFT data is structured into a chat (or message API) format during training, and only the loss on target tokens is computed to optimize the LLM's translation ability for incomplete text. However, during inference, it adopts a fixed policy (e.g., reading a fixed number of tokens at each step), leading to a mismatch between training and inference phases. In contrast, EAST introduces the training method the same as the pretraining, i.e., or next-token prediction (or completion API) format, to learn an adaptive read/write policy. Particularly, the loss is computed across source, target, and read/write tokens. This not only optimizes the translation performance but also enables LLMs to model adaptive read/write behaviors based on context, ensuring consistency between training and inference. As a result, EAST can effectively segment and translate source text into appropriate semantic units based on latency requirements specified by the instructions, achieving a latency-specific adaptive read/write policy.

B Data Statistics

The data statistics for our SiMT and OMT datasets are illustrated in Table 4 and 5, respectively.

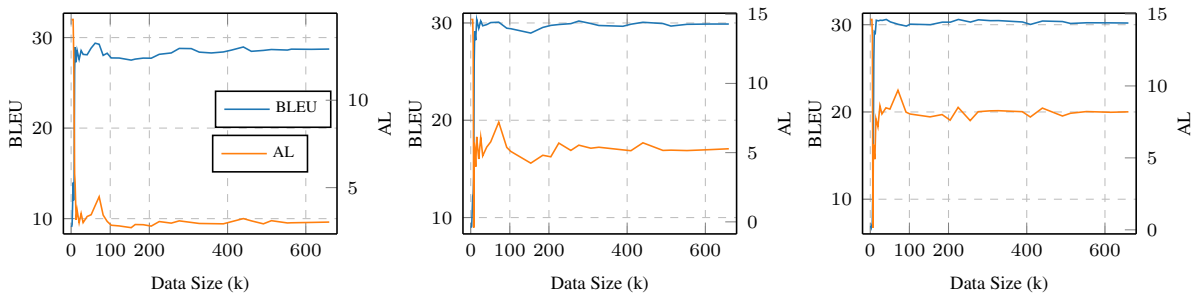
C Data processing

For each sentence pair from WMT15 De→En training set, we first filter out source sentences with less than 20 words. We then utilize LLMs to generate SiMT chunk sequences at three different latency levels, as outlined in Eq.(2), while filtering out invalid examples generated by LLMs with unequal numbers of source and target chunks. We found that such mismatches often result from non-monotonic translations. Next, we compute BLEURT scores between the LLM-generated translations and the ground-truth references, removing examples with scores below 80 to ensure translation quality. Additionally, we merge any chunk that contains fewer than two words (or four characters for Chinese) with its subsequent chunk to avoid overly short segments. This processing pipeline ensures that the final dataset is well-aligned and preserves monotonicity.

D Implementation Details

Our implementation is based on LLaMA-Factory[7] (Zheng et al., 2024). We train our models using Llama-3-8B-Instruct (Dubey et al., 2024) as the backbone, with full parameter tuning on Stage I and LoRA tuning on Stage II. All models are trained on 8 Nvidia A100 GPUs with a total batch size of 256, a learning rate of 1e-5, a cosine learning rate scheduler, a warm-up ratio of 0.1 and a maximum sequence length of 1024. The number of epochs is set to 1 for full parameter tuning and 2 for LoRA tuning, respectively. When using LoRA, the LoRA

[7] https://github.com/hiyouga/LLaMA-Factory



(a) Low Latency (b) Medium Latency (c) High Latency

Figure 7: BLEU scores (left y-axis) and AL values (right y-axis) over data size using normal scale.

Latency	SiMT-Multi-90K									SiMT-De-En-660K
	De→En	Zh→En	Ru→En	Cs→En	En→De	EN→Zh	En→Ru	En→Cs	Total	De→En
Low	3,325	1,635	3,642	2,507	3,267	2,423	4,945	4,226	25,970	230,902
Medium	3,631	2,763	3,719	2,472	3,997	2,433	5,830	5,035	29,880	227,131
High	4,102	4,166	4,254	2,746	4,921	2,920	6,322	5,433	34,864	202,843
Total	11,058	8,564	11,615	7,725	12,185	7,776	17,097	14,694	90,714	660,876

Table 4: The statistics for the two SiMT datasets we constructed.

rank, alpha, and dropout rate are set to 64, 128, and 0.05, respectively.

E Additional Results

E.1 COMET-AL and BLEURT-AL Curves for Main Results

In Figures 8 and 9, we present the COMET-AL and BLEURT-AL curves across multiple language pairs for the WMT22 X→En and En→X test sets.
COMET-AL Results: EAST achieves consistently superior COMET scores compared to both Conversational SimulMT and Llama3-MNMT with wait-k across all language pairs, especially in the low-latency region. The adaptive policy of EAST effectively balances latency (AL) and quality, achieving a higher COMET score.
BLEURT-AL Results: EAST demonstrates strong BLEURT performance across the language pairs and achieves the best BLEURT scores on average. However, in En-De, Llama3-MNMT w/ wait-k achieves comparable BLEURT scores.

E.2 How Different Training Strategies Affect Performance

We conduct comprehensive experiments between EAST and the following variants.

1. **EAST-Stage-I**: Full-weight fine-tuning on the SiMT-De-En-660K dataset.
2. **EAST-Single-Stage**: Full-weight fine-tuning on all the three datasets for one single epoch.
3. **EAST-w/o-Offline**: Removing the Off-Multi-120K datasets in Stage II.
4. **EAST-Only-Stage-II**: Removing the Stage I fine-tuning.

SiMT Performance The BLEU-AL curves of SiMT X→En tasks are illustrated in first row of Figures 10. Notably, the EAST-Stage-I model, which is obtained from tuning the 660K De→En SiMT data alone, shows reasonable performance under varying latency instructions in language pairs like Ru→En and Cs→En due to linguistic similarities among these Indo-European languages, facilitating better transfer learning. The Stage I model struggles with correct translation for Zh→En because of the significant structural and grammatical differences between Chinese and Indo-European languages. However, EAST with two-stage training greatly improves the performance of Zh→En becomes normal, underscoring the importance of this approach. Additionally, EAST with two-stage training further enhances performance across multiple latency ranges for De→En, Ru→En, and Cs→En, with improvements of 0.5 to 1 in BLEU.

The Stage I model completely fails to perform En→X translations, by repeating the source English sentence. This underperformance in En→X

Language	Sentence-level Parallel Data			Document-level Parallel Data		
	Train	Test (from En)	Test (to En)	Test (to English)	Avg. Words	Max. Words
German (De)	14211	2037	1984	217	107/123	839/1003
Chinese (Zh)	15406	2037	1875	-	-	-
Russia (Ru)	15000	2037	2016	128	175/211	699/843
Czech (Cs)	12076	2037	1448	-	-	-

Table 5: The statistics for the parallel data from the WMT. "Avg. Words" indicates the average number of words per document in the source/target language. "Max. Words" represents the maximum number of words per document in the source/target language.

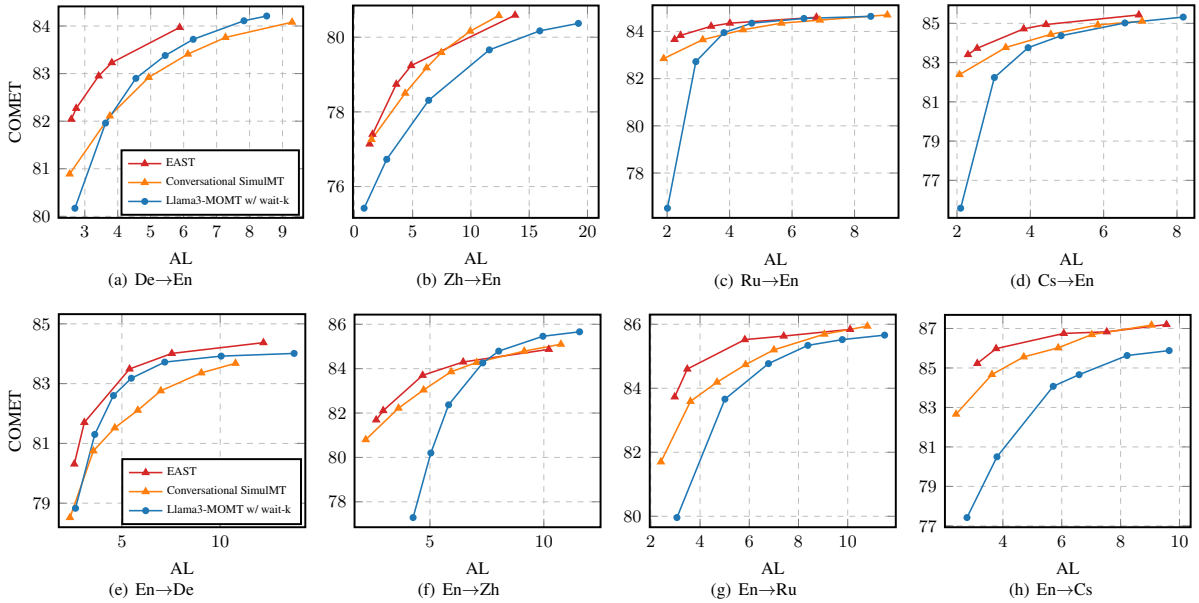


Figure 8: COMET-AL curves for main results on the WMT22 X→En and En→X test sets.

directions without specific fine-tuning on those language pairs highlights the challenges of cross-linguistic semantic structures in SiMT. Fortunately, a very smaller multilingual dataset with 90K SiMT parallel pairs enable the LLM excellent performance on the reversed language directions En→X. Compared with LLM-based SiMT baselines, EAST demonstrates superior performance across all 8 translation directions. In Figure 16 and 17 of the Appendix, we also plot the quality-latency curves of all methods with respect to COMET and BLEURT, revealing a trend similar to that of the BLEU-AL curves.

The EAST-w/o-Offline variant, which removes the OMT data in Stage II, shows a slight performance decline in De→En and Ru→En but maintained similar performance in other language pairs. The EAST-Only-Stage-II that omits the Stage I fine-tuning results in a performance degradation of about 1 BLEU for the X→En on average, whereas the translation performance remains relatively unchanged for En→X. This suggests that learning a novel SiMT task may require a larger scale dataset. When fine-tuning with all three high-quality datasets in a single stage, EAST-Single-Stage demonstrates competitive performance across various delays and language orientations. Specifically, it achieves even higher BLEU-AL curves than the full two-stage training pipeline for both En→De and En→Cs. However, the two-stage training approach offers the advantage of better generalization to novel language directions with a reduced training schedule, avoiding re-training on the extensive Stage I dataset.

Offline Performance We also evaluate the performance of offline translation on the WMT22 test set, as presented in Table 6. Compared to offline NMT model Llama3-MNMT and other variants, EAST maintained comparable or superior offline translation performance across the eight language directions, indicating that our two-stage SFT process effectively maintains translation quality for offline

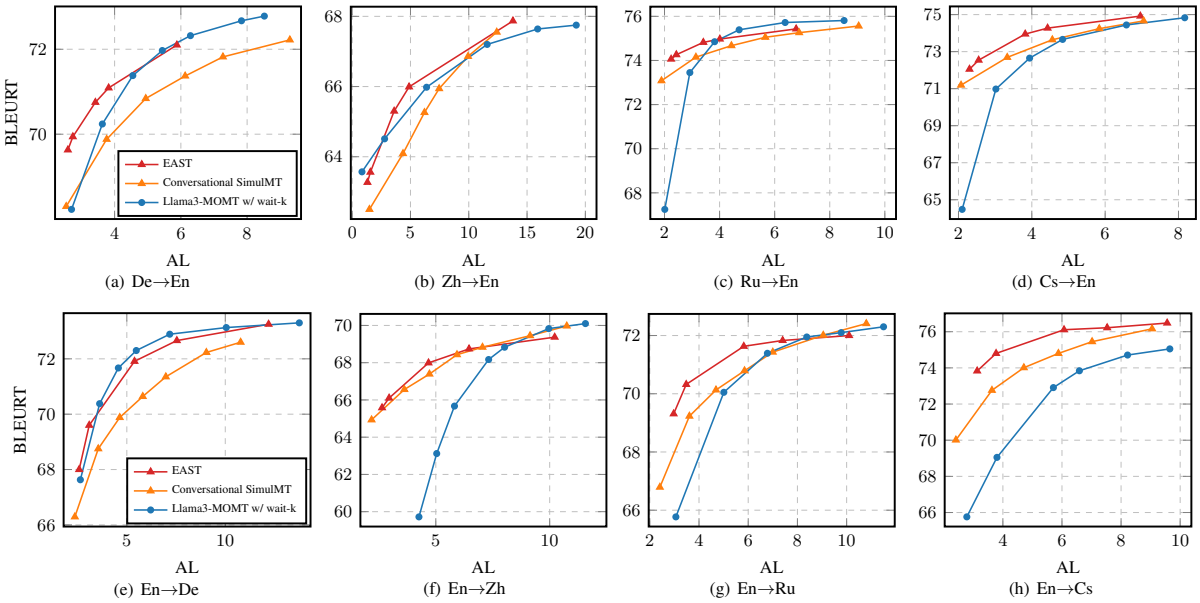


Figure 9: BLEURT-AL curves for main results on the WMT22 X→En and En→X test sets.

Models	De→En		Zh→En		Ru→En		Cs→En		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Llama3-MOMT	31.98	**84.89**	**25.48**	**81.26**	39.83	**85.19**	44.92	**86.23**	**35.55**	**84.39**
EAST	**32.55**	84.77	23.80	80.86	39.83	85.04	**45.61**	86.20	35.45	84.22
EAST-Single-Stage	30.01	84.15	24.05	80.20	36.06	84.39	39.12	84.63	32.31	83.34
EAST-w/o-Offline	32.37	84.55	22.42	80.85	**40.29**	84.80	41.21	85.41	34.07	83.90
EAST-Only-Stage-II	31.34	84.34	24.90	80.89	38.48	84.78	42.77	85.97	34.37	84.00

Models	En→De		En→Zh		En→Ru		En→Cs		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Llama3-MOMT	30.45	**85.63**	**40.68**	**86.53**	24.83	87.27	**27.92**	88.36	30.97	**86.95**
EAST	30.84	85.49	40.17	86.31	**26.79**	87.13	26.63	88.17	31.11	86.78
EAST-Single-Stage	**30.85**	85.51	39.69	86.43	26.57	87.32	27.76	**88.40**	**31.22**	86.92
EAST-w/o-Offline	26.77	84.34	28.27	84.69	23.17	85.89	23.27	87.00	25.37	85.48
EAST-Only-Stage-II	30.50	85.44	39.03	86.31	26.62	**87.40**	26.85	88.32	30.75	86.87

Table 6: Offline results for different training strategies on the WMT22 X→En and En→X test sets. **Bold** values denote the highest scores, while the underlined values indicate the second highest scores.

NMT. EAST-Single-Stage shows excellent translation performance for En→X, although it slightly underperforms by about 3 BLEU and 1 COMET for X→En. The EAST-w/o-Offline model, not even trained on offline translation data, still performed well, particularly for X→En. This can be attributed to the fact that our high-latency SiMT data has context that is close to being as informative as the offline NMT data. Similar to the trend in SiMT, the offline performance of the EAST-Only-Stage-II drops by 1 BLEU and 0.22 COMET for X→En, while the performance remains relatively stable for En→X. In summary, these results highlight EAST not only excels in high-quality simultaneous translation but also ensures that the offline translation capabilities are not compromised.

E.3 What Is hallucination Rate of The LLM-based SiMT?

Hallucination is a significant challenge in traditional SiMT, as the models begin translating while receiving input. This can prompt incorrect assumptions about the content yet to be received, resulting in hallucinated outputs. Additionally, hallucination is a common issue in the outputs of LLMs across various generation tasks. Therefore, it is more essential to evaluate the hallucination phenomenon in LLM-based SiMT. To effectively measure the hallucinations in our case, we utilize the hallucination rate (HR) metric (Chen et al., 2021), which quantifies the proportion of target words in the hypothesis that do not align with any source words.

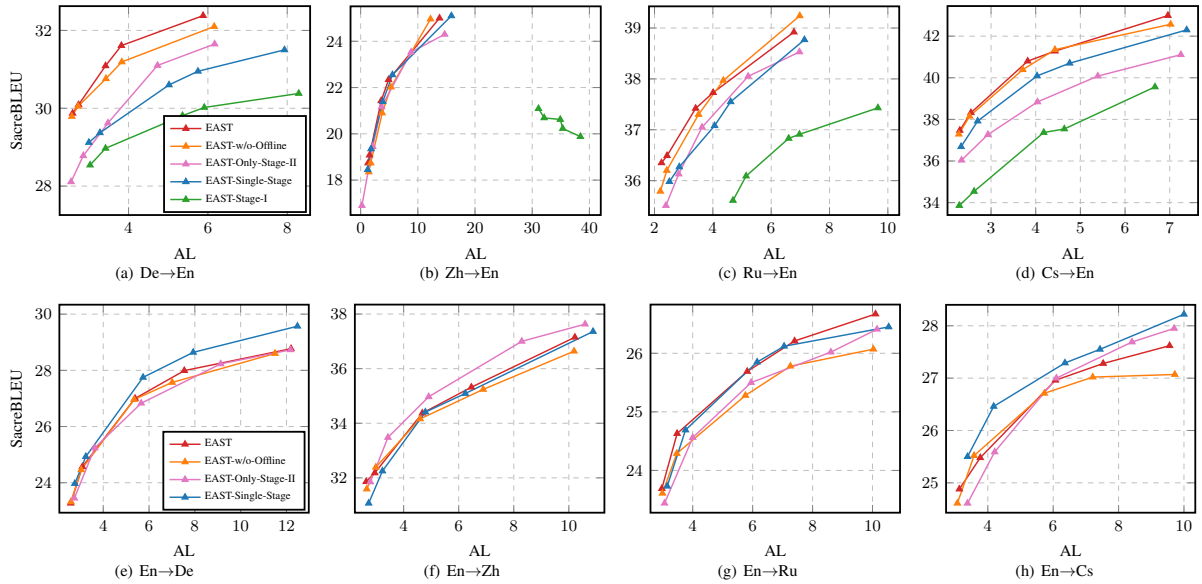


Figure 10: BLEU-AL curves for different training strategies on the WMT22 X→En and En→X test sets.

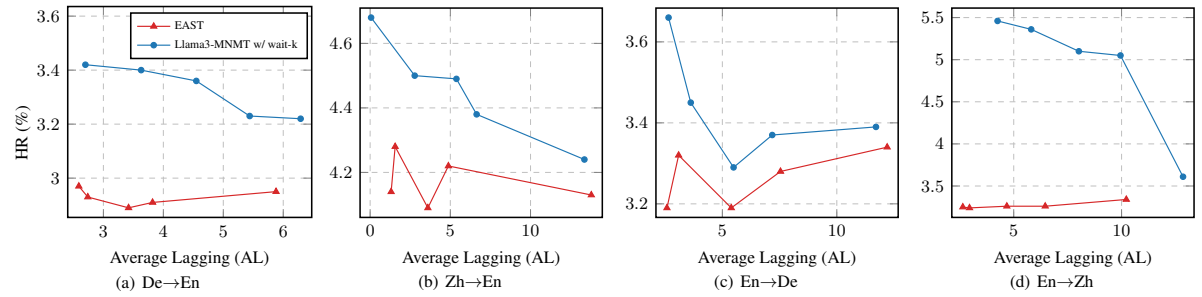


Figure 11: The hallucination rate (HR) against the latency metrics (AL) on the WMT22 test sets.

For this, we employ the fast-align[8] tool to identify word-level alignments between the source text and the target translation.

Figure 11 illustrates the HR comparison on En↔De and En↔Zh test sets. EAST consistently demonstrates a lower hallucination rate across all latency levels and test sets compared to the Llama3-MOMT w/ wait-k. Unlike the wait-k policy, EAST can adaptively determine reading and writing actions based on the semantic context. This prevents the model from prematurely generating translations, thereby reducing the production of hallucinated content and ensuring translations that are more accurate and faithful to the source text.

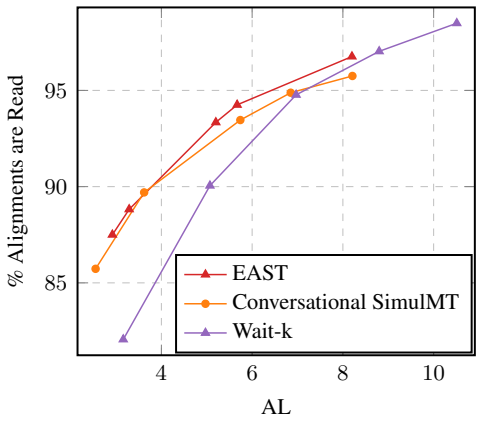


Figure 12: The proportion of the ground-truth aligned source tokens received before translating.

E.4 Quality of Translation Policy

To evaluate the quality of our translation policy, we conduct experiments on the manually aligned RWTH De→En alignment dataset[9]. Following (Zhang and Feng, 2022), we measure the proportion of ground-truth aligned source tokens that are

[8] https://github.com/clab/fast_align

[9] https://www-i6.informatik.rwth-aachen.de/goldAlignment/

read before generating each target token. Specifically, for a target token y_i, the number of source tokens read (g_i) must be at least equal to the ground-truth aligned source position (a_i). This ensures that the alignment between y_i and x_{a_i} is satisfied during the SiMT process. The proportion is calculated as follows:

$$\mathrm{A} = \frac{1}{T}\sum_{i=1}^{T}\mathbb{I}_{a_i \leq g_i} \tag{4}$$

where T is the total number of target tokens and $\mathbb{I}_{a_i \leq g_i}$ counts the number of $a_i \leq g_i$.

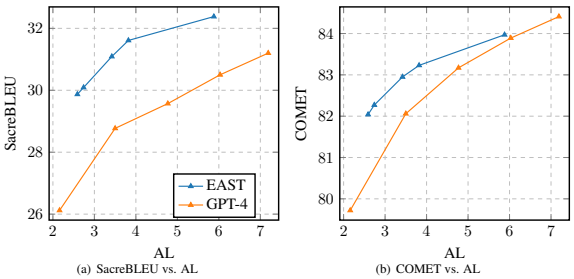
As shown in Figure 12, EAST consistently achieves the higher percentage of aligned source tokens read before translating across most latency levels compared to Conversational SimulMT and Wait-k. This result indicates that EAST better adheres to the ground-truth alignment, ensuring sufficient source context is read before generating target tokens.

E.5 Analysis on Different Backbone LLMs

In this section, we analyze the impact of different backbone models, Llama-3-8B-Instruct, Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct, on translation performance.

SiMT Performance As shown in Figure 15, EAST-Llama3.1 achieves similar or higher performance compared to EAST-Llama3 across all eight language pairs, indicating that a more powerful backbone model leads to better translation quality. The curves for EAST-Llama3.1 and EAST-Llama3 are generally higher than those for EAST-Qwen2.5, demonstrating the superiority of Llama-based models in multilingual SiMT tasks. However, for Zh→En, En→Zh and En→Ru, EAST-Qwen2.5 achieves better performance than both Llama models. This can be attributed to the pretraining data distribution, where Qwen2.5 likely benefits from having been pre-trained on a larger proportion of Chinese monolingual data, enhancing its performance on tasks involving the Chinese and Russian.

Offline Performance The performance trends observed in Table 8 align closely with those seen in SiMT. EAST-Llama3.1 consistently outperforms EAST-Llama3, while EAST-Qwen2.5 delivers competitive results only in Zh→En and En→Zh. Importantly, EAST models achieve translation performance comparable to their offline counterparts, demonstrating that the EAST framework effectively preserves high translation quality.



(a) SacreBLEU vs. AL (b) COMET vs. AL

Figure 13: Translation quality and latency results on the WMT22 De→En test set.

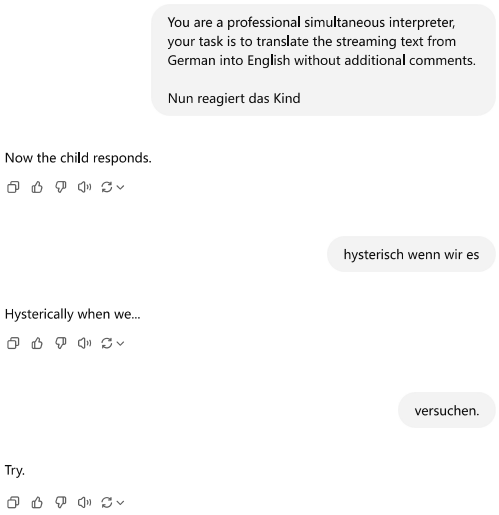


Figure 14: A SiMT case for GPT-4.

E.6 Comparison with GPT-4

GPT-4 demonstrates superior offline translation quality in Table 2, largely due to its extensive offline translation training corpus and larger model size. Since GPT-4 is a closed-source model, it cannot effectively execute adaptive SiMT as EAST does. Instead, we use a fixed-policy inference strategy similar to Conversational SimulMT, where a fixed number of words are read before GPT-4 generates the corresponding translation. The experimental results are shown in Figure 13. The results indicate that GPT-4 significantly underperforms EAST across all latency levels. A key reason for this performance gap is that GPT-4 has not been fine-tuned on SiMT-specific data, which can't efficiently perform zero-shot simultaneous translations. For example, when provided with partial source input, GPT-4 tends to add a period at the end of this response (Figure 14), leading to suboptimal translations and reduced performance.

Latency Level	Fluency (↑)	BLEU (↑)	AL (↓)
Low	7.25	28.27	2.47
Medium	7.55	30.60	4.53
High	7.81	32.44	9.44
Offline	8.29	33.28	18.80

Table 7: Translation fluency, BLEU, and AL of EAST under different latency levels. Fluency scores are rated by GPT-4 on a 0–10 scale.

E.7 Fluency Evaluation

Table 7 presents the fluency scores of translations generated by EAST under varying latency levels, as rated by GPT-4 on a 0–10 scale. The results are averaged over eight translation directions on the WMT22 test set (X→En and En→X). These results show that while offline translation achieves the highest fluency, our method maintains consistently high fluency across all latency levels, with only a minor drop compared to offline. This demonstrates that our method preserves fluency well.

E.8 Numeric Results for the Figures

We also provide the numeric results for Figures 3 in Tables 9 and for Figures 4, 8, and 9 in Tables 10.

F Prompt

The example for SiMT SFT data is shown in the Figure 18. The prompt template for generating SiMT data is provided in Figure 19. The instruction data for offline translation is shown in the Figure 20. The fluency evaluation prompt is provided in Figure 21.

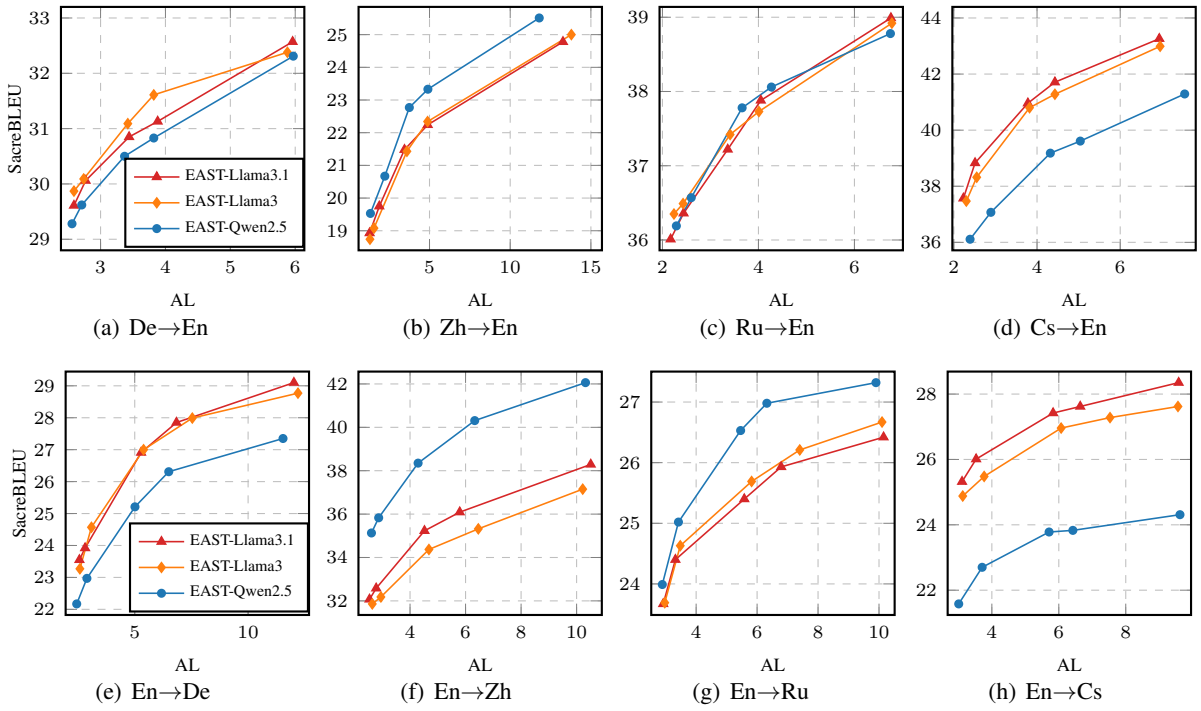


Figure 15: BLEU-AL curves for different backbone LLMs on the WMT22 X→En and En→X test sets.

Models	De→En		Zh→En		Ru→En		Cs→En		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Llama3-MOMT	31.98	84.89	25.48	81.26	39.83	85.19	44.92	86.23	35.55	84.39
Llama3.1-MOMT	31.80	**84.90**	26.87	81.46	**40.45**	85.42	**45.92**	**86.50**	**36.26**	**84.57**
Qwen2.5-MOMT	31.62	84.73	**27.33**	**81.78**	40.15	**85.60**	43.85	85.48	35.74	84.40
EAST-Llama3	32.55	84.77	23.80	80.86	39.83	85.04	45.61	86.20	35.45	84.22
EAST-Llama3.1	**32.62**	84.80	26.22	81.12	39.98	85.08	44.89	86.24	35.93	84.31
EAST-Qwen2.5	32.33	84.52	25.62	81.46	40.29	85.44	44.24	85.58	35.62	84.25

Models	En→De		En→Zh		En→Ru		En→Cs		Average	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Llama3-MOMT	30.45	85.63	40.68	86.53	24.83	87.27	27.92	88.36	30.97	86.95
Llama3.1-MOMT	**32.11**	**85.78**	40.65	86.70	27.28	**87.51**	**29.75**	**89.10**	**32.45**	**87.27**
Qwen2.5-MOMT	29.57	84.62	**43.88**	**87.56**	**27.86**	**87.51**	21.48	86.26	30.70	86.49
EAST-Llama3	30.84	85.49	40.17	86.31	26.79	87.13	26.63	88.17	31.11	86.78
EAST-Llama3.1	31.13	85.68	40.75	86.38	27.10	87.20	28.53	88.67	31.88	86.98
EAST-Qwen2.5	28.70	84.32	41.97	87.17	27.66	87.29	23.75	86.45	30.52	86.31

Table 8: Offline results for different backbone LLMs on the WMT22 X→En and En→X test sets.

Latency	EAST-Stage-I w/ Llama3				EAST-Stage-I w/ Llama2			
	AL	BLEU	COMET	BLEURT	AL	BLEU	COMET	BLEURT
Low	2.68	32.46	84.28	72.52	3.18	32.55	84.32	72.62
Low-Medium	3.06	33.12	84.63	72.90	3.46	32.80	84.51	72.88
Medium	4.77	35.21	85.66	74.29	5.84	34.62	85.49	74.28
Medium-High	5.33	35.55	85.72	74.47	7.05	34.93	85.70	74.57
High	7.78	36.62	86.11	75.02	9.73	35.49	85.91	74.83

Table 9: Numeric results on WMT15 De→En test set for EAST-Stage-I w/ Llama3 and EAST-Stage-I w/ Llama2 (Figure 3).

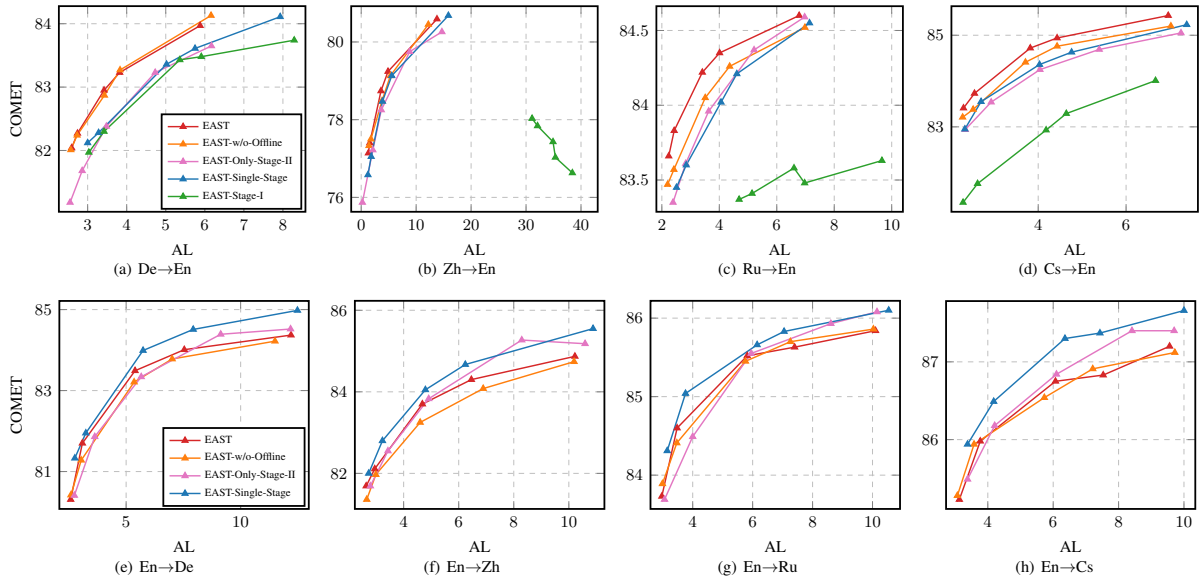


Figure 16: COMET-AL curves for different training strategies on the WMT22 X→En and En→X test sets.

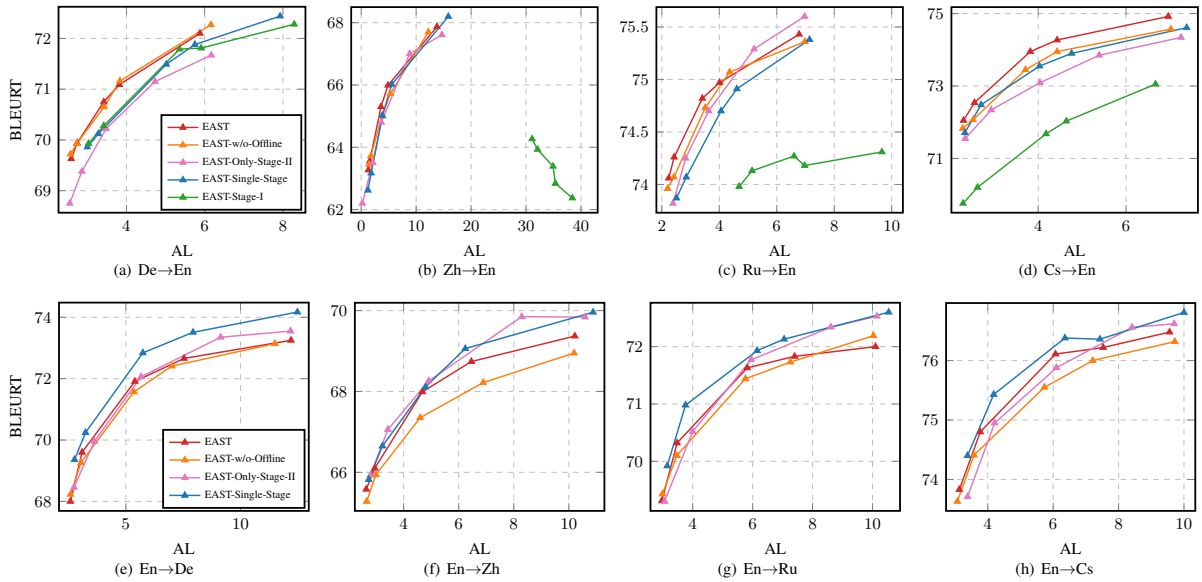


Figure 17: BLEURT-AL curves for different training strategies on the WMT22 X→En and En→X test sets.

<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful assistant.<|eot_id|><|start_header_id|>user<|end_header_id|>

Translate the following text from English into German with ***low*** *latency.* <|eot_id|>
<|start_header_id|>assistant<|end_header_id|>

Anyone with information<|end-of-read|> Jeder, der Informationen hat,<|end-of-write|> is asked to call<|end-of-read|> wird gebeten,<|end-of-write|> the SFPD Tip Line<|end-of-read|> das Hinweistelefon des SFPD<|end-of-write|> at 415-575-4444.<|end-of-read|> unter 415-575-4444 anzurufen.<|end-of-write|><|eot_id|>

Figure 18: An example of the SiMT SFT data for Llama-3. Prompt is colored in gray. The source and target texts are highlighted in cyan and orange, respectively. The read-write tokens are highlighted in blue and red, respectively. We calculate the loss for all tokens other than the prompt during training.

As a professional simultaneous interpreter, your task is to segment sentences into independent semantic chunks and provide corresponding English translations.

You will use three different granularities for segmentation:
1. For low latency, the chunks would be fragmented into brief, coherent phrases that convey a complete thought.
2. For medium latency, the chunks would be longer, possibly clause or sentence-long segments.
3. For high latency, the chunks would be the longest, likely to cover complete clauses or full sentences.

You also need to provide corresponding simultaneous translation for each segment by performing the translation monotonically while making the translation grammatically tolerable.

Please take into consideration the example attached below:

Input:
Chinese: 休斯敦16日晚发出一系列龙卷风和严重雷暴警报。

Output:
```
{
    "low_latency": {
        "Chinese": ["休斯敦", "16日晚", "发出一系列", "龙卷风", "和严重雷暴", "警报。"],
        "English": ["Houston", "on the evening of the 16th", "issued a series of", "tornado", "and severe thunderstorm", "warnings."]
    },
    "medium_latency":{
        "Chinese": ["休斯敦16日晚", "发出一系列", "龙卷风和严重雷暴警报。"],
        "English": ["On the evening of the 16th, Houston", "issued a series of", "tornado and severe thunderstorm warnings."]
    },
    "high_latency": {
        "Chinese": ["休斯敦16日晚", "发出一系列龙卷风和严重雷暴警报。"],
        "English": ["On the evening of the 16th, Houston", "issued a series of tornado and severe thunderstorm warnings."]
    }
}
```

Figure 19: The prompt template for GPT-4 to generate SiMT data.

<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful assistant.<|eot_id|><|start_header_id|>user<|end_header_id|>

Translate the following text from English into German.

Anyone with information is asked to call the SFPD Tip Line at 415-575-4444. <|eot_id|>

<|start_header_id|>assistant<|end_header_id|>

Jeder, der Informationen hat wird gebeten das Hinweistelefon des SFPD unter 415-575-4444 anzurufen.

<|eot_id|>

Figure 20: An example of the OMT SFT data for Llama-3. The source and target texts are highlighted in cyan and orange, respectively. We compute the loss on the target tokens during training.

Fluency Evaluation

Evaluate the fluency of the translated text based on the following scoring criteria from 0 to 10, with a minimum granularity of 1 point.

Scoring Criteria:

- 10: Perfectly matches the target language's expression habits, with no grammatical, spelling, punctuation, or word order issues. The language is idiomatic and reads as if written by a native speaker.
- 8: Mostly natural and fluent, with only minor grammar or word usage issues. The reader can easily understand the entire text and almost won't notice any unnaturalness.
- 6: The translation is somewhat stilted, with several grammar or word usage issues. Some sentences require the reader to adjust their understanding. Overall, the original meaning can still be understood.
- 4: Poor fluency, with frequent grammatical and expression errors. Reading is laborious but the general idea can still be understood.
- 2: Extremely stilted, with most content being hard to understand. The language is disorganized and almost fails to convey the message.
- 0: Completely incomprehensible.

Please output the score in the following JSON format: {"fluency": "score" }

Translated text:

Figure 21: Prompt template for fluency evaluation.

Latency	De→En				En→De			
	AL	BLEU	COMET	BLEURT	AL	BLEU	COMET	BLEURT
Low	2.59	29.87	82.04	69.63	2.58	23.27	80.31	68.00
Low-Medium	2.74	30.09	82.27	69.94	3.09	24.57	81.70	69.60
Medium	3.42	31.09	82.95	70.75	5.39	27.00	83.49	71.91
Medium-High	3.82	31.61	83.23	71.09	7.54	27.99	84.01	72.66
High	5.88	32.38	83.97	72.10	12.20	28.77	84.37	73.25

Latency	Zh→En				En→Zh			
	AL	BLEU	COMET	BLEURT	AL	BLEU	COMET	BLEURT
Low	1.31	18.74	77.14	63.27	2.64	31.86	81.69	65.58
Low-Medium	1.56	19.08	77.40	63.56	2.95	32.18	82.11	66.10
Medium	3.60	21.43	78.74	65.30	4.68	34.37	83.70	67.99
Medium-High	4.88	22.34	79.24	65.99	6.46	35.32	84.30	68.74
High	13.79	25.00	80.59	67.87	10.22	37.15	84.87	69.37

Latency	Ru→En				En→Ru			
	AL	BLEU	COMET	BLEURT	AL	BLEU	COMET	BLEURT
Low	2.24	36.35	83.66	74.06	2.97	23.69	83.73	69.31
Low-Medium	2.43	36.49	83.83	74.26	3.48	24.63	84.60	70.32
Medium	3.41	37.42	84.22	74.82	5.82	25.69	85.52	71.63
Medium-High	4.01	37.73	84.35	74.97	7.40	26.21	85.63	71.83
High	6.78	38.92	84.60	75.43	10.10	26.67	85.84	72.00

Latency	Cs→En				En→Cs			
	AL	BLEU	COMET	BLEURT	AL	BLEU	COMET	BLEURT
Low	2.30	37.47	83.41	72.05	3.13	24.88	85.23	73.83
Low-Medium	2.55	38.32	83.73	72.54	3.77	25.48	85.98	74.80
Medium	3.82	40.80	84.72	73.95	6.07	26.96	86.75	76.11
Medium-High	4.43	41.28	84.94	74.27	7.53	27.28	86.83	76.22
High	6.96	42.99	85.43	74.92	9.56	27.62	87.20	76.48

Table 10: Numeric results on WMT22 X→En and En→X test sets for EAST (Figures 4, 8, and 9).