# LLMs Are Zero-Shot Context-Aware Simultaneous Translators

**Roman Koshkin**[†] **Katsuhito Sudoh**[‡♣] **Satoshi Nakamura**[‡♠]
[†]Okinawa Institute of Science and Tenchnology, Japan
[‡]Nara Institute of Science and Technology, Japan
[♠]The Chinese University of Hong Kong, Shenzhen
[♣]Nara Women's University, Japan
roman.koshkin@oist.jp

## Abstract

The advent of transformers has fueled progress in machine translation. More recently large language models (LLMs) have come to the spotlight thanks to their generality and strong performance in a wide range of language tasks, including translation. Here we show that open-source LLMs perform on par with or better than some state-of-the-art baselines in simultaneous machine translation (SiMT) tasks, zero-shot. We also demonstrate that injection of minimal background information, which is easy with an LLM, brings further performance gains, especially on challenging technical subject-matter. This highlights LLMs' potential for building next generation of massively multilingual, context-aware and terminologically accurate SiMT systems that require no resource-intensive training or fine-tuning. The code is available at https://github.com/RomanKoshkin/toLLMatch.

## 1 Introduction

In simultaneous translation, the translator – either a machine or human – is expected to start the translation *before* the source sentence is finished, often making strong assumptions about the meaning of certain words, phrases, or the intent of the entire message. To produce a coherent – although not necessarily accurate – translation, human simultaneous translators routinely use a range of techniques, one of which is delaying the translation of an initially ambiguous word or phrase in the hope that its meaning will become resolved by later context (Ilyukhin, 2001; Chernov, 2004; Setton, 2005; Amos et al., 2022). Perhaps more importantly, human translators reduce this inherent uncertainty by relying on information from other sources, such as presentation slides and glossaries of standard terms. This, and the fact that some people insist on using the term "interpreter", rather than "translator"[1], highlights a very different nature of this kind of translation.

Despite significant progress in the field of offline machine translation, recently enabled by the wide adoption of the transformer architecture (Vaswani et al., 2017), the practical use of SiMT systems is still limited due to a range of unsolved problems. One of these problems is that existing SiMT systems – in stark contrast to human simultaneous translators – operate on a sentence level, completely disregarding the context established by previous sentences, or the broader (extralinguistic) context that is implied, but not contained in the text itself. Needless to say, such context-unaware translation is often logically incoherent and is prone to terminological inconsistencies, especially across long discourse. The very fact that human interpreters – even the most experienced professionals – routinely prepare for upcoming translation jobs by studying relevant subject-matter, reviewing or compiling topic-specific glossaries of terms, names, and job titles (Álvarez Pérez and Pérez-Luzardo Díaz, 2022; Gile, 1986, 1985; Chernov, 1978), suggests that SiMT systems should have access to *additional* information needed to make terminologically appropriate and accurate translation.

Motivated by LLMs' strong reasoning (Yao et al., 2023; Huang et al., 2024; Huang and Chang, 2023; Zhou et al., 2024), translation (Xu et al., 2024; Zhu et al., 2024) and in-context learning (Liu et al., 2022; Wei et al., 2022; Brown et al., 2020) capabilities, we attempt to address one of the weaknesses of existing SiMT systems, namely that their translation takes no account of the wider context and generally cannot respect specific terminological constraints. Different from previous studies which have attempted fine-tuning LLMs for SiMT tasks (Wang et al., 2023; Agostinelli et al., 2024;

[1]Following the practice established in the machine translation community, in this paper we will be using the term "simultaneous translation".



*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1207

Koshkin et al., 2024), our focus here is on translation in zero-shot mode. In the method we propose, the LLM receives a prompt that contains *both* the partial input, partial translation and minimal background information, and generates the next word of the translation. At the next step, the prompt is updated with the new source and the newly translated word (see Section 3 for details). We show empirically that such an approach outperforms some of the strongest bilingual SiMT baselines and shows competitive results to a state-of-the-art multilingual SiMT system. Importantly, our approach makes it easy to insert background information (see Fig. 1 and Section 4), which helps the LLM to make *contextually appropriate* word choices.

Our key contributions are as follows:

1. We show that an off-the-shelf instruction-tuned LLM can successfully perform a SiMT task zero-shot, without a sophisticated segmentation policy, with quality and latency metrics that are competitive with (and in some cases exceeding) the state of the art.

2. We show that instruction-tuned LLMs can be easily used for contextually-aware SiMT, and that injecting *minimal* background information generally improves the quality of the translation by a large margin.

3. We propose *response priming*, which consists in fixing the initial part of the assistant's response, and improves the LLM's zero-shot performance on SiMT tasks.

The rest of the paper is structured as follows. In Section 2 we provide an overview of recent SiMT literature. In Section 3 we describe our method and the datasets used for evaluating our method. In Section 4 we demonstrate the performance of our approach on the different datasets and language pairs. We conclude with a discussion of limitations and future directions and in Section 5.

## 2 Related work

Simultaneous machine translation (SiMT) systems strive to balance translation quality – commonly evaluated using the BLEU metric (Papineni et al., 2002) – with acceptable latency levels. This balance is managed through a "policy" that determines the timing of translation actions (i.e., a WRITE action) versus the reception of additional input (i.e., a READ action). The literature classifies these policies into two main types: fixed and adaptive (Zhang et al., 2020). Fixed policies, such as *wait-k* (Ma et al., 2019), apply predefined rules for executing READ and WRITE actions, regardless of the textual context. Initially, SiMT models employed *chunk-based* strategies (Bangalore et al., 2012; Yarmohammadi et al., 2013; Fügen et al., 2007; Sridhar et al., 2013), where the text is divided into sub-sentence segments for translation without considering the context from preceding chunks, leading to reduced translation accuracy. In response to these drawbacks, Dalvi et al. (2018) introduced an *incremental decoding* method. This technique enhances chunk translations by integrating preceding contexts via the hidden states of an RNN. Paired with straightforward segmentation tactics, their method surpassed the performance of prior state-of-the-art systems. Meanwhile, adaptive policies, such as "wait-if" rules (Cho and Esipova, 2016), allow for more flexible WRITE/READ actions by considering parts of the source and/or target text. Adaptive policies can be developed using separately trained agents, often employing reinforcement learning techniques (Alinejad et al., 2018; Satija and Pineau, 2016; Grissom II et al., 2014; Gu et al., 2017). These policies may initiate READ/WRITE actions based on model attention mechanisms (Ma et al., 2020; Arivazhagan et al., 2019; Raffel et al., 2017; Chiu and Raffel, 2018) or the stability of output predictions across $n$ steps, a concept referred to as "local agreement" (Polák et al., 2022; Ko et al., 2023; Liu et al., 2020a). Recent research has also investigated policy training using binary search strategies (Guo et al., 2023) to optimize the translation quality improvement per token processed, and has conceptualized the translation actions as a hidden Markov transformer (Zhang and Feng, 2023), where hidden events indicate optimal translation output times.

A promising area of research, related to this study, focuses on adapting encoder-decoder transformers like mBART (Liu et al., 2020b), initially developed for sentence-level translation, to the SiMT task. Significant advances have been made in multilingual translation models (Fan et al., 2020; Tang et al., 2020), with some work focusing on creating more efficient versions of large models (Mohammadshahi et al., 2022). For instance, Kano et al. (2022); Fukuda et al. (2023) have applied fine-tuning techniques using prefix-alignment data, while Zhang et al. (2020) have employed fine-

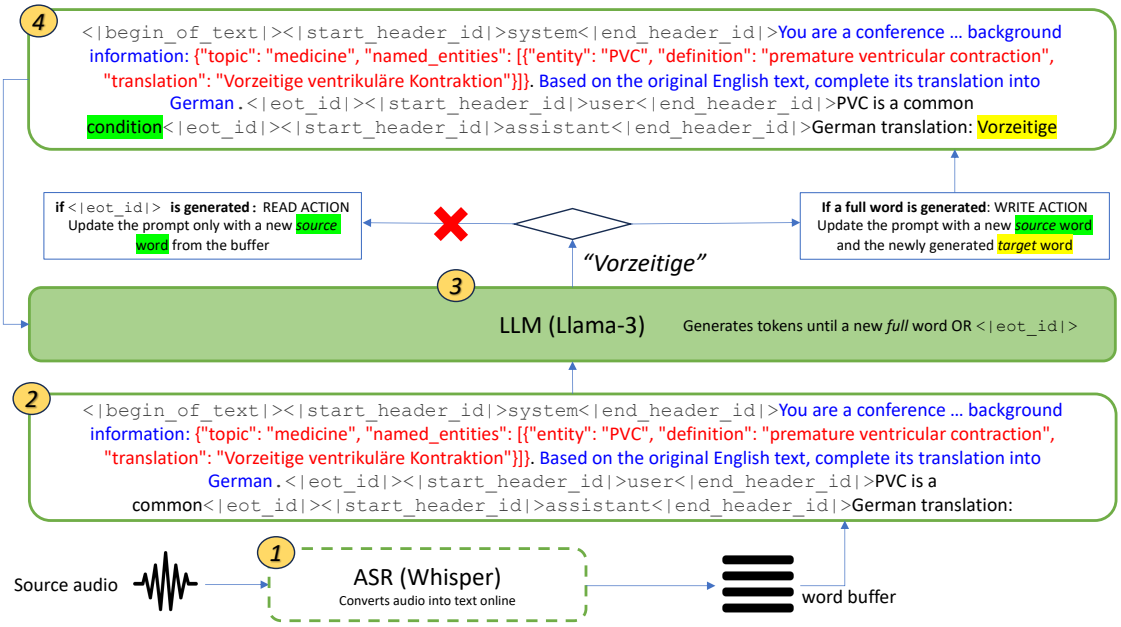


Figure 1: Model overview. Chunks of input audio are incrementally processed by WHISPER (1), and the recognized words are stored in the buffer. The prompt (2) includes special strings (shown in grey), system message (blue) with background information (red) to constrain the space of possible translations, and the model's previous translation (if exists). Given the prompt, the LLM's generates tokens until either a new full word or <|eot_id|> is generated (3). If a new full word is generated, a WRITE action is performed: a new source word from the word buffer and the newly generated word ("Vorzeitige" in this example) are added to the prompt. If <|eot_id|> is generated, a READ action is performed: the prompt is updated only with a new source word from the buffer.

tuning on "meaningful units", both demonstrating strong performance across various language pairs.

More recently, large language models (LLMs) have demonstrated remarkable capabilities across a wide range of tasks, including offline machine translation (Xu et al., 2024; Zhu et al., 2024). Importantly, LLMs' ability to learn in-context enables a range of new capabilities, such as terminology-constrained translation (Moslem et al., 2023) and self-correction of translation errors (Feng et al., 2024). These and other developments raised the question whether LLMs can be leveraged for SiMT. Recent works have explored various ways to fine-tune LLMs for SiMT and showed that coupled with a segmentation policy, such as wait-k (Wang et al., 2023) or more sophisticated "local agreement" (Agostinelli et al., 2024), it can deliver competitive performance on some language pairs. Koshkin et al. (2024) proposed a policy-free approach, in which an LLM is fine-tuned on pairs of "causally aligned" source-target sentence pairs to act as both the translator and segmentation policy at the same time.

Distinct from previous literature, we show that an off-the-shelf instruction-tuned LLM can perform SiMT zero-shot, eliminating the need for resource-intensive model training and the complexities of making special datasets and fine-tuning. Importantly, our approach enables *context-aware SiMT* which, as we empirically demonstrate, substantially improves translation quality.

## 3 Method

### 3.1 Online ASR

Similarly to Koshkin et al. (2024), we follow a cascaded approach, where an automatic speech recognition (ASR) model (WHISPER (Radford et al., 2023)) incrementally converts input audio chunks into text which is fed into the LLM for translation. We found that for English input `whisper-small.en`[2] achieved approximately the same word error rate (WER) of about about 5% as `whisper-large-v3`, so we chose the smaller version for faster inference. Although trained on full sentences, WHISPER can still perform online ASR with the following simple technique. For each

[2] https://huggingface.co/openai/whisper-small.en

READ action, a new segment of audio, lasting 200 ms, is added to any previously read audio chunks and then processed by WHISPER. This window length was chosen empirically as a trade-off between, on the one hand, the desire to minimize translation latency and word error rate (WER): larger windows typically are likely to result in lower WER, but tend to increase latency metrics. In our online ASR, we discard the last predicted word unless the entire source audio has been read in.

Similarly to Koshkin et al. (2024), the output of the ASR cascade is fed into the LLM (`Llama-3-70B-Instruct`[3]). However, in an important distinction from Koshkin et al. (2024), we insert the partial target not into the "user", but the "assistant" part of the prompt (Fig. 1). This simple modification, which we call *response priming*, effectively limits the space of possible sequences that the model can produce and prevents it from generating apologies, explanatory notes or other undesirable additions to the translation.

### 3.2 Evaluation Data

For the English-German language pair we used FLEURS (Conneau et al., 2023) and TED-TST-2023 (Koshkin et al., 2024). However, it is possible that those test sets (or the data that they were built from) were leaked into the LLM's pre-training set. For this reason we created another dataset – which we call TED-TST-2024 – similar in size and content type to TED-TST-2023, but only including talks posted after the LLM was released.

```
{
    "topic": "Climate Crisis and Fossil Fuel Industry's
         Influence",
    "named_entities":
        [
            {"entity": "troposphere",
            "description": "the lowest part of the
                 atmosphere"},
            {"entity": "Inflation Reduction Act",
            "description": "U.S. legislation aimed at
                 addressing climate change"},
            {"entity": "COP process",
            "description": "Conference of the Parties,
                 climate change conferences"},
            {"entity": "COP28",
            "description": "upcoming climate conference
                 hosted by UAE"},
        ]
}
```

Listing 1: Example of background information used to augment TED-TST-2023 and TED-TST-2024.

Additionally, to showcase the ability of LLMs to leverage background information for improved SiMT, we context-augment TED-TST-2023 and TED-TST-2024 with relevant background information (Listing 1).

We generated this background information with `gpt-4-turbo-2024-04-09` by prompting it with the entire TED talk for which a given sentence was taken (the full prompt is in Appendix A). The idea here is to make the translation more realistic by providing the translator (the LLM in our case) with essential information about the subject-matter at hand.

Finally, we test our model in a more challenging scenario imitating translation of highly technical subject-matter. Prior to translating complex, technical subject matter, human interpreters compile topic-specific glossaries, which typically list terms from the source language along with their definitions and standard translations into the target language (Álvarez Pérez and Pérez-Luzardo Díaz, 2022; Gile, 1986, 1985; Chernov, 1978). This preparatory work is crucial for effectively conveying technical content, as it equips interpreters with the precise terminology and contextual knowledge needed to handle subject-specific nuances. Motivated by this, we constructed AMBIEVAL, which is a context-augmented dataset of ambiguous terms, which we describe next. First we collect a list of English words (some of which are acronyms) that can have very different meanings in different contexts. For example, depending on the context, the word "MOS" can mean "metal oxide semiconductor" and also "military occupational specialty". Sometimes, the meaning of the word is disambiguated later in the sentence. Consider the following two examples:

*One must watch out for kicks, which are dangerous influxes of formation fluids into the wellbore.*

*One must watch out for kicks, while maintaining a strong defense and executing effective strikes.*

In these sentences, the meaning of the word "kicks" is disambiguated by later context, specifically by the words "influxes" and "strikes". Unless background information is somehow fed into the model together with the source, it is difficult for the SiMT model to immediately translate the word "kicks" accurately. We also create examples with words whose meaning cannot be disambiguated based on the information contained within the sentence, for example:

*The CPA recommends holding pharmaceutical companies to stricter standards of accountability.*

In this sentence, "CPA" is never disambiguated

[3]At the time of writing this paper, Meta had released the 8B and 70B versions of the model, but not the corresponding paper or technical report.

and can mean almost anything (e.g. "Consumer Protection Act", "Canadian Psychiatric Association", "Cerebral Palsy Alliance"). The source audio of AMBIEVAL is generated by Amazon's Polly text-to-speech service.

### 3.3 Inference

For inference, we follow a similar approach to TRANSLLAMA (Koshkin et al., 2024), but also inject background information. Specifically, at time $t$, the target token $y_t$ is conditional on all the source tokens $x_{\leq t}$ revealed up to time $t$, previously generated target tokens $x_{<t}$ and background information $b$, which is constant for sentences coming from the same text (speech).

$$p(y_t|y_{<t}, x_{\leq t}, b) \quad (1)$$

Given a prompt (Fig. 2) consisting of a system message, partial input and previously translated partial target, the LLM greedily generates one or more new tokens. Once a new full word is generated, a WRITE action is performed. A READ action is performed when an <|eot_id|> token is generated. A WRITE action involves adding the next source word and the newly translated target word to the prompt. In a READ action, the prompt is only updated by inserting the next source word into the prompt. WRITE actions are only permitted after the length of the input audio reaches a certain minimum length. This constraint controls latency-quality trade-off and indirectly the WER: higher values of this minimum length generally improve the quality by increasing the average number of words the LLM gets at the beginning of translation and decreasing the WER of the ASR[4]. Except for the temperature (set to 0 for greedy generation), all the generation parameters were left at their default values.

After all the source words have been revealed, the input is no longer partial and no new words are added to it, but the generation process continues until <EOS>. We illustrate the inference process in Fig. 1 and Algorithm 1.

For fast inference, we use the `vllm`[5] library which implements a range of latest LLM performance optimizations, most importantly tensor parallelism. Unless otherwise noted, all the results reported in this paper were obtained on a Linux machine with 4 A100 80GB GPUs. The ASR cascade was run using `whisper-jax`[6] an implementation of WHISPER built for maximum inference speed.

**Algorithm 1** Inference process

```
partial_output = []

# do ASR after MIN_T s of audio is read
asr = ASR(min_t=MIN_T)
llm = LLM()

while True:
    # get the next audio chunk, recognize
    (partial_input,
    audio_finished) = asr.next()

    prompt = " ".join([
        SYSTEM_MSG,
        background_info,
        partial_input,
        partial_output])

    # generate until full word
    # or '<|eot_id|>'
    next_word = llm.generate(prompt)

    if next_word == "<|eot_id|>":
        if audio_finished:
            break    # finish sentence
        else:
            continue # READ
    else:
        # WRITE
        partial_out.append(
            next_word)
```

### 3.4 Prompt structure

We follow a similar prompt structure as in Koshkin et al. (2024) (Fig. 2), except that we do not instruct the LLM to generate special <WAIT> tokens, but inject background information as part of the system message. For the SYSTEM_MESSAGE we used the following text: *"You are a conference interpreter. As you translate, you can use the following background information:* BACKGROUND_INFORMATION_JSON. *Taking into account the original* SRC_LANG *text, complete its translation into* TGT_LANG. *Do not add any notes or comments to the translation."* This system message performed well empirically, and we speculate that further improvements are possible with different system messages. We leave this question to future work.

[4] if the initial audio segment is too short, WHISPER is more likely to hallucinate words that were never said.

[5] https://github.com/vllm-project/vllm

[6] https://github.com/sanchit-gandhi/whisper-jax

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>
SYSTEM_MESSAGE
BACKGROUND_INFORMATION_JSON
USER_INSTRUCTION
<|eot_id|><|start_header_id|>user<|end_header_id|>
Context: PARTIAL_SOURCE
<|eot_id|><|start_header_id|>assistant<|end_header_id|>
German translation: PARTIAL_TARGET
```

Figure 2: Prompt structure. `<|begin_of_text|>`, `<|start_header_id|>ROLE_NAME<|end_header_id|>`, and `<|eot_id|>` are special strings used in Llama-3 to flank the system, user and assistant parts of the the prompt.

## 4 Results

### 4.1 Benchmarks

In this section we compare the performance of our method to SEAMLESSSTREAMING (Barrault et al., 2023), which is a state-of-the-art massively multilingual SiMT system on five language pairs (en-{de,es,fr,it,ru}) and additionally to three recent bilingual SiMT systems, namely: NAIST (Fukuda et al., 2023), FBK (Papi et al., 2023) and TRANSLLAMA[7] (Koshkin et al., 2024) on the en-de pair.

We start by examining the quality-latency trade-off on TED-TST-2024 (Fig. 3). Our method performed strongly relative to the recent baselines (although not on all language pairs). In all of the results presented in this section, we controlled the translation latency by varying the minimum length of the audio before allowing WRITE actions (OURS and TRANSLLAMA), attention threshold (SEAMLESSSTREMING and FBK) and source segment size (NAIST).

| Method | BLEU | AL | LAAL |
|---|---|---|---|
| Ours | 22.13 | 1360.59 | 2089.16 |
| NAIST | 21.39 | 1060.94 | 1967.36 |
| FBK | 17.65 | 1645.42 | 1922.79 |
| SEAMLESS | 19.75 | 1442.71 | 1781.06 |
| TRANSLLAMA | 19.36 | 1732.08 | 2017.91 |

Table 1: Quality (BLEU (Papineni et al., 2002)) and latency (average lagging (AL) (Ma et al., 2019) and length-adaptive average lagging (LAAL) (Papi et al., 2022)) for our approach compared with state-of-the-art baselines on the en-de language pair on TED-TST-2023.

[7]We used the version of TRANSLLAMA derived from Llama-2-70B.

| Method | BLEU | AL | LAAL |
|---|---|---|---|
| Ours | 32.30 | 1720.00 | 2022.05 |
| NAIST | 36.44 | 1615.80 | 2120.09 |
| SEAMLESS | 31.75 | 1695.24 | 1877.11 |
| FBK | 15.56 | 1744.59 | 2028.93 |
| TRANSLLAMA | 25.71 | 1820.33 | 2095.07 |

Table 2: Quality and latency results for our approach compared with state-of-the-art baselines on the en-de language pair on FLEURS.

When benchmarking our model against the baselines on the FLEURS, TED-TST-2023, and AMBIEVAL datasets, we approximately matched the length-aware average lagging (LAAL) (Papi et al., 2022) to 2000 ms.

| Method | BLEU | AL | LAAL |
|---|---|---|---|
| Ours | 42.60 | 1961.57 | 2008.48 |
| FBK | 24.96 | 1906.59 | 2151.32 |
| NAIST | 39.80 | 1662.06 | 1796.68 |
| SEAMLESS | 29.76 | 1937.35 | 1978.72 |
| TRANSLLAMA | 32.43 | 1838.81 | 1903.21 |

Table 3: Quality and latency results for our approach compared with state-of-the-art baselines on the en-de language pair on AMBIEVAL.

Additional performance tests on TED-TST-2023 (Table 1) and FLEURS (Table 2) further demonstrate the performance of our approach. Since TED-TST-2023 and TED-TST-2024 are built from content intended for lay audiences, and therefore is relatively easy to translate, we also evaluate our method on another dataset (AMBIEVAL) which models a more challenging scenario where the meaning of some technical terms cannot be resolved immediately or without additional contex-

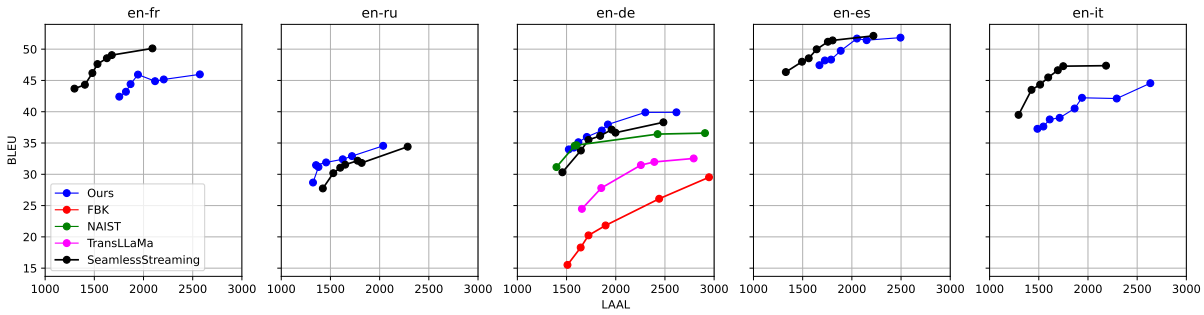


Figure 3: Dependence of translation quality (measured by BLEU) on latency (measured by LAAL) for `en-{fr, ru, de, es, it}` on TED-TST-2024. The latency was controlled by varying the minimum length of the audio before allowing WRITE actions (OURS and TRANSLLAMA), attention threshold (SEAMLESSSTREMING and FBK) and source segment size (NAIST).

tual information (see Section 3.2). As expected, our method outperforms the baselines by a large margin (Table 3, but also see Section 4.4).

### 4.2 Inference speed

One might wonder if using an LLM for real-time SiMT is feasible in practice. While our system has much more parameters than the state-of-the-art SiMT baselines (except for TRANSLLAMA), it can still achieve real-time translation if run on a modern inference engine that leverages a range of optimizations such as tensor parallelism (Table 4).

| Method | bn params | RTF |
|---|---|---|
| Ours | 70.79 | 0.86 |
| NAIST | 1.04 | 1.34 |
| FBK | 0.176 | 0.42 |
| SEAMLESS | 1.96 | 0.36 |
| TRANSLLAMA | 70.52[8] | 15.3 |

Table 4: Parameter counts and real-time factor (RTF) of the chosen baselines and our model. See Appendix D for information about model hyperparameters and how RTF was calculated.

As long as the entire system – including the ASR cascade and LLM – can function with an RTF of 1 or less, it can in principle be used for live simultaneous translation.

### 4.3 Recovery from ASR errors

Beyond the ability to ingest additional (background) information, another advantage of LLM-based translation is the ability to recover from ASR errors (Chen et al., 2023; Hu et al., 2024; Yang et al., 2023; Ma et al., 2023). Although on the TED datasets WHISPER produces a very low WER ($< 5\%$), these errors might still negatively impact the translation quality. Inspection of the translated texts reveals that compared to a state-of-the art offline translation model (NLLB-200 (NLLB Team et al., 2022)) Llama-3 is very good at correcting ASR errors, for example:

**ASR output:** *I think terrorists like Hamas and his bala are evil, and there is a bright line between groups that aim to kill innocence and those that try to avoid doing so at all costs.*

**LLM translation:** *Ich denke , Terroristen wie Hamas und Hezbollah sind böse, und es gibt eine klare Grenze zwischen Gruppen, die unschuldige Menschen töten wollen, und jenen, die alles tun, um dies zu vermeiden.*

**NLLB translation:** *Ich denke, Terroristen wie die Hamas und seine Bala sind böse, und es gibt eine klare Linie zwischen Gruppen, die Unschuld töten wollen, und denen, die versuchen, dies um jeden Preis zu vermeiden.*

In the example above, two ASR errors (underlined in the ASR output) were corrected by the LLM, but not by NLLB-200. For more examples, see Appendix B.

### 4.4 Ablations

**Response priming**. Table 5 shows that removing response priming from the prompt results in a small but consistent decrease of translation quality. This makes sense because response priming constrains the space of possible sequences that the LLM can generate in response to the prompt. Inspection of the translations revealed that without response priming the translations often begin with unwanted notes, comments and explanations resulting in decreased quality.

[8] Assuming `whisper-large-v2` is used for ASR.

| priming | en-de | en-es | en-fr | en-it | en-ru |
|---|---|---|---|---|---|
| yes | 41.43 | 54.87 | 47.21 | 40.24 | 36.38 |
| no | 39.52 | 54.53 | 46.06 | 38.71 | 36.11 |

Table 5: Disabling response priming consistently decreases translation quality across all the five language pairs. The numbers are mean BLEU scores over five runs with different latencies on TED-TST-2024.

**Background information**. The removal of minimal background information notably decreases the translation quality (Table 6), highlighting that the LLM can leverage even minimal information for improved quality. Notably, the smaler version of LLAMA-3 does not seem to benefit from added background information (Table 7), which is likely due to the fact that smaller LLMs generally have weaker instruction-following and in-context learning abilities.

| background | en-de | en-es | en-fr | en-it | en-ru |
|---|---|---|---|---|---|
| no | 31.14 | 46.04 | 41.76 | 36.38 | 29.11 |
| yes | 36.76 | 49.81 | 44.57 | 40.26 | 31.87 |

Table 6: Removing background information from the prompt significantly and consistently decreases quality across the all the five language pairs. The numbers are mean BLEU scores over five runs with different latencies on TED-TST-2024.

**Smaller LLMs**. Is is possible to achieve comparable performance (in terms of quality) with a smaller LLM? Our tests show that, unfortunately, `Meta-Llama-3-8B-Instruct` significantly underperforms its larger version, `Meta-Llama-3-70B-Instruct` and seems to be unable to benefit from background information (Table 7). Inspection of the translations suggests that the the smaller LLM is much worse at exactly following the instruction to only output the translation and nothing else.

## 5 Limitations and Future Directions

Prior work has demonstrated that fine-tuning on a small dataset is sufficient to enable an LLM to perform the challenging task of simultaneous translation. However, these existing approaches are potentially limited to one language pair, involve constructing a specialized dataset and a non-trivial search for optimal fine-tuning hyperparameters. Here we demonstrate the an off-the-shelf instruction-tuned LLM performs strongly zero-shot on several different datasets and, crucially, can leverage additional information for improved quality and/or adherence to a predefined list of technical terms, which is important in translating technical material.

| background | pair | BLEU | AL | LAAL |
|---|---|---|---|---|
| yes | en-de | 30.52 | 2311.31 | 2466.86 |
| | en-fr | 41.91 | 2609.47 | 2678.53 |
| | en-es | 41.76 | 2520.15 | 2626.96 |
| | en-ru | 26.14 | 2018.06 | 2254.75 |
| | en-it | 31.76 | 2356.28 | 2567.44 |
| no | en-de | 30.42 | 2313.28 | 2404.13 |
| | en-fr | 41.96 | 2621.87 | 2691.50 |
| | en-es | 42.79 | 2519.84 | 2605.44 |
| | en-ru | 26.40 | 2025.78 | 2226.59 |
| | en-it | 36.23 | 2357.07 | 2454.95 |

Table 7: A smaller LLM performs significantly worse than the default 70B version. Results are shown for the TED-TST-2024 dataset.

In the future, as stronger and more lightweight models become available, the LLM can analyze its own translations and/or summarize source sentences or paragraphs. These summaries could be added to a vector store or a graph database and retrieved in real time to augment the translation of future sentences.

The big performance gap between the 8B and 70B version of LLAMA-3 suggests that even better translation quality could be achieved with larger closed-source models (such as GPT-4 or CLAUDE) if their APIs allowed response priming.

One practical limitation of our approach is that currently, to the best of our knowledge, it cannot be used with strong closed-source models that are available through API. Perhaps as a countermeasure against model jailbreaking, the APIs through which these instruction-tuned models (e.g. GPT-4, Claude and Gemini) can be accessed enforce a rigid prompt structure that is incompatible with *response priming* – specifying a user-specified prefix for the (assistant) model's response – which is at the core of our approach.

Another significant bottleneck in our LLM-based simultaneous translation system is that it relies on a separate ASR system that was not designed for online operation. Although in general this cascaded setup works well, hallucinations sometimes occur, especially in low-latency regimes when in response to initial silence WHISPER outputs words that were never said in the audio. We believe this limitation can be addressed by implementing an end-to-end SiMT system, in which the

output embeddings of an ASR system or speech encoder would be directly projected into the LLM's input embedding space, bypassing a text representation and improving the system's latency overall. In fact, there is already some work in this direction, e.g. by Fathullah et al. (2024) and Huang et al. (2023).

It is interesting to explore other ways to improve the performance and efficiency of our method, such as local agreement (Polák et al., 2022), efficient weight quantization (e.g. `awq` (Lin et al., 2024)), and more sophisticated prompting strategies.

## Acknowledgements

The first author acknowledges financial support from KAKENHI grant JP23KJ2131 and Google.

## 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.

Ashkan Alinejad, Maryam Siahbani, and Anoop Sarkar. 2018. Prediction improves simultaneous neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3022–3027, Brussels, Belgium. Association for Computational Linguistics.

Beneharo Álvarez Pérez and Jessica María Pérez-Luzardo Díaz. 2022. Interpreter preparation in the interpreting classroom environment. a study on the usefulness of terminological glossaries. *Interpreters Newsletter*.

Rhona M. Amos, Kilian G. Seeber, and Martin J. Pickering. 2022. Prediction during simultaneous interpreting: Evidence from the visual-world paradigm. *Cognition*, 220:104987.

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. 2019. Monotonic infinite lookback attention for simultaneous machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1313–1323, Florence, Italy. Association for Computational Linguistics.

Srinivas Bangalore, Vivek Kumar Rangarajan Sridhar, Prakash Kolan, Ladan Golipour, and Aura Jimenez. 2012. Real-time incremental speech-to-speech translation of dialogs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 437–445.

Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, et al. 2023. Seamless: Multilingual expressive and streaming speech translation. *arXiv preprint arXiv:2312.05187*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chen Chen, Yuchen Hu, Chao-Han Huck Yang, Sabato Marco Siniscalchi, Pin-Yu Chen, and Eng-Siong Chng. 2023. Hyporadise: An open baseline for generative speech recognition with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 31665–31688. Curran Associates, Inc.

Ghelly V Chernov. 2004. Inference and anticipation in simultaneous interpreting. *Amsterdam and Philadelphia: Benjamins*.

G.V. Chernov. 1978. *Theory and Practice of Simultaneous Interpretation*. International Relations.

Chung-Cheng Chiu and Colin Raffel. 2018. Monotonic chunkwise attention. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Kyunghyun Cho and Masha Esipova. 2016. Can neural machine translation do simultaneous translation? *arXiv preprint arXiv:1606.02012*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, and Stephan Vogel. 2018. Incremental decoding and training methods for simultaneous translation in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 493–499, New Orleans, Louisiana. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Çelebi, Guillaume Wenzek,

Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22:107:1–107:48.

Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Junteng Jia, Yuan Shangguan, Ke Li, Jinxi Guo, Wenhan Xiong, Jay Mahadeokar, Ozlem Kalinli, Christian Fuegen, and Mike Seltzer. 2024. Prompting large language models with speech recognition abilities. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13351–13355.

Zhaopeng Feng, Yan Zhang, Hao Li, Wenqiang Liu, Jun Lang, Yang Feng, Jian Wu, and Zuozhu Liu. 2024. Improving llm-based machine translation with systematic self-correction.

Christian Fügen, Alex Waibel, and Muntsin Kolss. 2007. Simultaneous translation of lectures and speeches. *Machine translation*, 21:209–252.

Ryo Fukuda, Yuta Nishikawa, Yasumasa Kano, Yuka Ko, Tomoya Yanagita, Kosuke Doi, Mana Makinae, Sakriani Sakti, Katsuhito Sudoh, and Satoshi Nakamura. 2023. NAIST simultaneous speech-to-speech translation system for IWSLT 2023. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 330–340, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Daniel Gile. 1985. Les termes techniques en interprétation simultanée. *Meta*, 30(3):199–210.

Daniel Gile. 1986. Le travail terminologique en interprétation de conférence.

Alvin Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. 2014. Don't until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1342–1352, Doha, Qatar. 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. 2023. Learning optimal policy for simultaneous machine translation via binary search. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2318–2333, Toronto, Canada. Association for Computational Linguistics.

Yuchen Hu, Chen Chen, Chengwei Qin, Qiushi Zhu, Eng Siong Chng, and Ruizhe Li. 2024. Listen again and choose the right answer: A new paradigm for automatic speech recognition with large language models. In *Annual Meeting of the Association for Computational Linguistics*.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1049–1065, Toronto, Canada. Association for Computational Linguistics.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Zhichao Huang, Rong Ye, Tom Ko, Qianqian Dong, Shanbo Cheng, Mingxuan Wang, and Hang Li. 2023. Speech translation with large language models: An industrial practice. *ArXiv*, abs/2312.13585.

Vladimir Mikhailovich Ilyukhin. 2001. *Strategies in Simultaneous Interpreting: Based on the Material of English-Russian and Russian-English Combinations*. Candidate of philological sciences dissertation, Moscow. Specialty 10.02.20: Comparative-Historical, Typological, and Comparative Linguistics.

Yasumasa Kano, Katsuhito Sudoh, and Satoshi Nakamura. 2022. Simultaneous neural machine translation with prefix alignment. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 22–31, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Yuka Ko, Ryo Fukuda, Yuta Nishikawa, Yasumasa Kano, Katsuhito Sudoh, and Satoshi Nakamura. 2023. Tagged end-to-end simultaneous speech translation training using simultaneous interpretation data. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 363–375, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Roman Koshkin, Katsuhito Sudoh, and Satoshi Nakamura. 2024. Transllama: Llm-based simultaneous translation system. *ArXiv*, abs/2402.04636.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for llm compression and acceleration. In *MLSys*.

Danni Liu, Gerasimos Spanakis, and Jan Niehues. 2020a. Low-Latency Sequence-to-Sequence Speech Recognition and Translation by Partial Hypothesis Selection. In *Proc. Interspeech 2020*, pages 3620–3624.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020b. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

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.

Rao Ma, Mengjie Qian, Potsawee Manakul, Mark John Francis Gales, and Kate Knill. 2023. Can generative large language models perform asr error correction? *ArXiv*, abs/2307.04172.

Xutai Ma, Juan Miguel Pino, James Cross, Liezl Puzon, and Jiatao Gu. 2020. Monotonic multihead attention. In *International Conference on Learning Representations*.

Alireza Mohammadshahi, Vassilina Nikoulina, Alexandre Berard, Caroline Brun, James Henderson, and Laurent Besacier. 2022. SMaLL-100: Introducing shallow multilingual machine translation model for low-resource languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8348–8359, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. 2023. Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland. European Association for Machine Translation.

Team NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

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.

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

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

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

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

Colin Raffel, Minh-Thang Luong, Peter J. Liu, Ron J. Weiss, and Douglas Eck. 2017. Online and linear-time attention by enforcing monotonic alignments. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 2837–2846. JMLR.org.

Harsh Satija and Joelle Pineau. 2016. Simultaneous machine translation using deep reinforcement learning. In *ICML 2016 Workshop on Abstraction in Reinforcement Learning*.

Robin Setton. 2005. *Pointing to contexts: A relevance-theoretic approach to assessing quality and difficulty in interpreting*, volume 7. Walter de Gruyter Berlin/New York.

Vivek Kumar Rangarajan Sridhar, John Chen, Srinivas Bangalore, Andrej Ljolje, and Rathinavelu Chengalvarayan. 2013. Segmentation strategies for streaming speech translation. In *Proceedings of the 2013*

*Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 230–238.

Y. Tang, C. Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *ArXiv*, abs/2008.00401.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

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

Chao-Han Huck Yang, Yile Gu, Yi-Chieh Liu, Shalini Ghosh, Ivan Bulyko, and Andreas Stolcke. 2023. Generative speech recognition error correction with large language models and task-activating prompting. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc.

Mahsa Yarmohammadi, Vivek Kumar Rangarajan Sridhar, Srinivas Bangalore, and Baskaran Sankaran. 2013. Incremental segmentation and decoding strategies for simultaneous translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1032–1036.

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

Pei Zhou, Jay Pujara, Xiang Ren, Xinyun Chen, Heng-Tze Cheng, Quoc V. Le, Ed Huai hsin Chi, Denny Zhou, Swaroop Mishra, and Huaixiu Steven Zheng. 2024. Self-discover: Large language models self-compose reasoning structures. *ArXiv*, abs/2402.03620.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

# Appendix

## A Prompts

Prompt used to extract background information for TED-TST-2023 and TED-TST-2024:

```
Please extract the topic and named entities (which are
      either proper names, technical terms or acronyms) from
      the following text, and return them as a JSON object
      with the following fields: topic, named_entities({
      entity, description}). For example:

{
  "topic": "Climate Crisis and Fossil Fuel Industry's
       Influence",
  "named_entities": [
    {
      "entity": "troposphere",
      "description": "the lowest part of the atmosphere"
    },
    {
      "entity": "gravity-measuring satellite",
      "description": "satellite used to observe water
           surpluses and deficits"
    },
    {
      "entity": "Inflation Reduction Act",
      "description": "U.S. legislation aimed at addressing
           climate change"
    },
    {
      "entity": "fossil fuel industry",
      "description": "industry opposing climate legislation"
    },
    {
      "entity": "UN Secretary General",
      "description": "stated fossil fuel industry is the '
           polluted heart' of climate crisis"
    },
    {
      "entity": "COP process",
      "description": "Conference of the Parties, climate
           change conferences"
    },
    {
      "entity": "COP28",
      "description": "upcoming climate conference hosted by
           UAE"
    },
    {
      "entity": "Sultan Al Jaber",
      "description": "CEO of Abu Dhabi National Oil Company
           and president of COP28"
    },
    {
      "entity": "Paris Agreement",
      "description": "international treaty on climate change
           "
    },
    {
      "entity": "Chevron",
      "description": "company with 93% worthless and junk
           offsets"
    },
    {
      "entity": "World Bank",
      "description": "institution needing reform to aid
           developing countries"
    },
    {
      "entity": "gigafactory",
      "description": "large factory for producing batteries
           and renewable energy components"
    },
    {
      "entity": "Drew Shindell",
      "description": "researcher on CO2 reduction timelines"
    }
  ]
}

TEXT
```

## B Examples of ASR error correction

*Example 1*

**ASR output:** Our mistaken belief that the best leaders are those who generate the most noise, action, and sensational activity in the most traumatic circumstances.

**LLM translation:** Unsere irrige Annahme, dass die besten Führer jene sind, die den meisten Lärm, die meisten Aktionen und die spektakulärsten Aktivitäten in den dramatischsten Umständen erzeugen.

**Offline MT (NLLB):** Unsere falsche Überzeugung, dass die besten Führungskräfte sind, die die meisten Lärm, Aktion und sensationelle Aktivität in den traumatischsten Umständen erzeugen.

*Example 2*

**ASR output:** In contrast, the expeditions of the role Amundsen make for boring reading.

**LLM translation:** Im Gegensatz dazu stehen die Expeditionen des Roald Amundsen für langweilige Lektüre.

**Offline MT (NLLB):** Im Gegensatz dazu sind die Expeditionen der Rolle Amundsen für langweiliges Lesen.

*Example 3*

**ASR output:** If Shackleton is the swimmer who rushes recklessly into the water without understanding the currents or its own capabilities, amundsen is the swimmer who has spent a lifetime, humbly studying the river before entering the water in just the right spot, at just the right time and so makes it look easy.

**LLM translation:** Wenn Shackleton der Schwimmer ist, der sich ohne vorheriges Verständnis der Strömungen oder seiner eigenen Fähigkeiten tollkühn ins Wasser stürzt, dann ist Amundsen der Schwimmer, der ein Leben lang bescheiden den Fluss studiert hat, bevor er genau am richtigen Punkt und zur richtigen Zeit ins Wasser geht und es dadurch leicht aussehen lässt.

## C Examples of translation

English source: I would like to invite you on a little thought experiment.

German translation: Ich möchte Sie einladen zu einer kleinen Gedankenübung.

Russian translation: Я хотел бы пригласить вас на маленький мысленный эксперимент.

Italian translation: Vorrei invitarti a una piccola riflessione immaginativa.

French translation: Je voudrais vous inviter à une petite expérience de pensée.

Spanish translation: Me gustaría invitarle a una pequeña experiencia de pensamiento.

English source: Let's pretend that we're going on a polar expedition together, all of you and me, and we need to hire a captain.

German translation: Lassen Sie uns so tun, als würden wir gemeinsam eine Polarexpedition unternehmen, alle zusammen, Sie und ich, und wir müssen einen Kapitän einstellen.

Russian translation: Давайте представим, что мы отправляемся в полярную экспедицию вместе, все вы и я, и нам нужно нанять капитана.

Italian translation: Immaginiamo di essere in una spedizione polare insieme, tutti voi e io, e dobbiamo assumere un capitano.

French translation: Imaginons que nous partions ensemble pour une expédition polaire, vous et moi, et que nous devions embaucher un capitaine.

Spanish translation: Imaginemos que estamos embarcándonos en una expedición polar juntos, todos ustedes y yo, y necesitamos contratar a un capitán.

English source: One comes from a man who has already successfully achieved all four of the major polar goals, the North Pole and the South Pole, and the Northeast and the Northwest Passage.

German translation: Einer stammt von einem Mann, der bereits alles vier erfolgreich erreicht hat, was die großen polaren Ziele betrifft, den Nordpol und den Südpol sowie die Nordost- und die Nordwestpassage.

Russian translation: Один от мужчины, который уже успешно достиг все четырех полярных целей: Северного полюса и Южного полюса, а также Северо-восточного и Северо-Западного проходов.

Italian translation: Uno proviene da un uomo che ha già conseguito con successo tutti e quattro gli obiettivi polari maggiori, il Polo Nord e il Polo Sud e il Passaggio del Nordest e il Passaggio del Nordovest.

French translation: L'un vient de l'homme qui a déjà réussi à tout accomplir dans les quatre grands objectifs polaires, le Pôle Nord et le Pôle Sud, ainsi que le Passage du Nord-Est et le Passage du Nord-Ouest.

Spanish translation: Uno proviene de un hombre que ya ha logrado con éxito los cuatro objetivos polares principales, el Polo Norte y el Polo Sur , así como el Paso del Noreste y el Paso del Noroeste.

English source: In fact, three of these, he was the first person to accomplish.

German translation: Tatsächlich drei von ihnen war er der Erste, der dies erreicht hat.

Russian translation: Фактически, три из них он был первым человеком, который это совершил.

Italian translation: In realtà, tre di questi, fu la prima persona a realizzare.

French translation: En réalité, trois d'entre eux, il fut le premier à accomplir.

Spanish translation: De hecho, tres de ellos, fue la primera persona en lograr.

English source: Candidate B is a man who set off for the Antarctic four times, three times as the man in charge, and every time resulted in failure, catastrophe, or death.

German translation: Kandidat B ist ein Mann, der aufbrach, um den Südpol viermal zu erreichen, drei Mal landete er als Leiter und jedes Mal endete es in Misserfolg, Katastrophe oder Tod.

Russian translation: Кандидат Б - мужчина , который отправился в путь к Антарктике четыре раза, три раза это был он , кто руководил, и каждый раз это заканчивалось неудачей, катастрофой или смертью.

Italian translation: Candidato B è un uomo che partì per l'Antartico quattro volte, tre delle quali fu l'uomo al comando, e ogni volta il risultato fu un fallimento, una catastrofe o la morte.

French translation: Candidat B est un homme qui a entrepris une expédition vers l'Antarctique à quatre reprises, trois fois il était à la tête de l'expédition, et chaque fois cela s'est soldé par un échec, une catastrophe ou la mort.

Spanish translation: El candidato B es un hombre que partió hacia la Antártida cuatro veces, tres veces como hombre a cargo, y cada vez resultó en un fracaso, una catástrofe o la muerte.

English source: But in reality, we often trick ourselves into hiring Candidate B or someone like him.

German translation: Aber, in Wirklichkeit, tun wir uns oft selbst einen Gefallen, indem wir Kandidat B oder jemanden wie ihn einstellen.

Russian translation: Но, на самом деле, мы часто обманываем самих себя, нанимая кандидата Б или кого-то вроде него.

Italian translation: Ma, in realtà, spesso inganniamo noi stessi nell'assumere candidati come B o qualcuno simile a lui.

French translation: Mais, en réalité, nous trompons souvent nous-mêmes en embauchant le candidat B ou quelqu'un de semblable.

Spanish translation: Pero, en realidad, a menudo nos engañamos al contratar al candidato B o a alguien como él.

English source: Meanwhile, Candidate A, the Norwegian Roald Amundsen, by any metric, the most successful polar explorer to have ever lived, has been largely forgotten.

German translation: Inzwischen, Kandidat A, der Norweger, ähnlich wie Amundsen, nach jeder Messlatte, der erfolgreichste Polarforscher , der je gelebt hat, wurde größtenteils vergessen.

Russian translation: Между тем, кандидат А, норвежец Роальд Амундсен, по любому критерию, самый успешный полярный исследователь , когда-либо существовавший , был в значительной степени забыт.

Italian translation: Nel frattempo, il candidato A, il norvegese Roald Amundsen, secondo ogni parametro, il più grande esploratore polare di tutti i tempi, è stato largamente dimenticato.

French translation: Pendant ce temps, le candidat A, le Norvégien Roald Amundsen, selon tous les critères, l'explorateur polaire le plus réussi de tous les temps, est largement tombé dans l'oubli.

Spanish translation: Mientras tanto, el candidato A, el noruego Amundsen, según cualquier métrica, el explorador polar más exitoso que haya vivido jamás, ha sido en gran medida olvidado.

English source: I did a quick search in my university's library catalog before this talk, and I found no fewer than 26 books that celebrate Shackleton's leadership qualities.

German translation: Ich habe eine schnelle Suche im Bibliothekskatalog meiner Universität durchgeführt, bevor ich hierher kam, und fand nicht weniger als 26 Bücher, die Shackletons Führungsqualitäten feiern.

Russian translation: Я быстро поискал в библиотеке нашего университета перед этим докладом, и я нашел ни меньше , чем 26 книг, которые прославляют лидерство Шеклтона.

Italian translation: Ho fatto una ricerca rapida nel catalogo della biblioteca universitaria prima di questo intervento e ho trovato non meno di 26 libri che celebrano le qualità di leadership di Shackleton.

French translation: J'ai fait une recherche rapide dans le catalogue de la bibliothèque de mon université avant cette conférence, et j'ai trouvé pas moins de 26 livres qui célébrent les qualités de leadership de Shackleton.

Spanish translation: Hice una búsqueda rápida en el catálogo de la biblioteca de mi universidad antes de esta charla y encontré no menos de 26 libros que celebran las cualidades de liderazgo de Shackleton.

## D Details on calculating the RTF

The RTF values reported in Table 4 were obtained by running our model on TED-TST-2-2024 with the parameter settings needed to achieve a LAAL of approximately 2000. Specifically:

**NAIST**

```
source-segment-size 600-950
la-n 2
beam 5
```

**FBK**

```
extract-attn-from-layer 3
frame-num 2
attn-threshold 0.2-0.4
```

**SeamlessStreaming**

```
source-segment-size 400
decision-threshold 0.6-0.9
```

**TransLLaMa**

```
wait-k 1
min-read-time 1.2-1.8
```

```
asr-model whisper-large-v2
```

**Ours**

```
min-read-time 1.2-1.8
asr-model whisper-small.en
```

All the runs were on the same hardware as mentioned in the main text. The RTF was computed as the ratio of (wall) time it took the model to complete translation of the given dataset to the total duration of the corresponding source audio clips.

## E Dataset statistics

| Dataset name | $N$ |
|---|---|
| TED-TST-2023[9] | 102 |
| TED-TST-2024 | 478 |
| FLEURS[10] | 642 |
| AMBIEVAL | 96 |

Table 8: Number of samples ($N$).

[9] https://github.com/RomanKoshkin/transllama

[10] https://huggingface.co/datasets/google/fleurs/blob/main/data/en_us/test.tsv