Unlocking Reasoning Capability on Machine Translation in Large Language Models

Anonymous ACL submission

Abstract

Reasoning-oriented large language models (RLMs) achieve strong gains on tasks such as mathematics and coding by generating explicit intermediate reasoning. However, their impact on machine translation (MT) remains underexplored. We systematically evaluate several RLMs on the WMT24++ benchmark and find that enabling explicit reasoning consistently degrades translation quality across languages and models. Our structural analysis shows that MT reasoning traces are highly linear, lacking revision, self-correction, and exploration of alternative translations, which limits their usefulness. Controlled reasoning-injection experiments demonstrate that providing high-quality reasoning traces from stronger models does not reliably improve weaker models' performance, indicating the failure lies within the format of MT reasoning. To address this mismatch, we propose a structured reasoning framework tailored to translation, based on multi-step improvements, and dynamic iterative revision over difficult segments. Post-training a 111B RLM on such structured reasoning traces yields consistent gains over standard translation fine-tuning and injected generic reasoning baselines. Our findings demonstrate that reasoning must be task-structured to benefit MT.

1 Introduction

Reasoning-oriented language models (RLMs), that is, large language models (LLMs) trained to spend more compute generating intermediate thinking tokens before answering, have recently delivered large performance gains on reasoning-heavy benchmarks, with especially strong improvements in mathematics and coding tasks (Kimi Team, 2025; Mistral-AI, 2025; Qwen Team, 2025; NVIDIA, 2025). This paradigm is increasingly viewed as a general strategy for improving model reliability, controllability, and performance via test-time compute scaling, making it a central design axis for the next generation of foundation models (Wang et al., 2025a,c). As a result, these improvements have sparked interest in whether explicit reasoning can also help on other challenging generative tasks, including machine translation (MT).

At the same time, relative to the depth of work on math and programming evaluations, MT has seen fewer systematic studies of RLMs and reasoning-style prompting as a first-class design axis (Liu et al., 2025; Li et al., 2026; Zebaze et al., 2025), leaving open whether reasoning-based scaling transfers to high-stakes generative tasks.

This work bridges the gap by investigating whether and how reasoning capabilities contribute to machine translation performance. We begin by evaluating a diverse set of RLMs on the WMT24++ benchmark (Deutsch et al., 2025), comparing RLMs performance with and without thinking tokens. Contrary to findings on most reasoning-demanding tasks (Jiang et al., 2025; Sun et al., 2024), and contrary to the common assumption that explicit reasoning is broadly beneficial across domains, our results show that enabling explicit reasoning does not improve translation quality. In fact, we observe a decline in performance when RLMs generate reasoning traces for MT.

To better understand this behavior, we conduct a qualitative analysis of the generated reasoning trajectories in Section 3.1. We find that these traces exhibit a linear structure lacking exploration of alternative translations, self-correction, and revision. This contrasts with math and code problems, where effective reasoning involves hypothesis exploration, backtracking, and verification (Marjanovic et al., 2026; Cuesta-Ramirez et al., 2025), behaviors that align well with search-based problem structures but appear mismatched with the incremental, constraint-driven nature of translation.

We further examine whether reasoning traces generated by a stronger RLM can benefit a weaker model, enhancing the performance. More specif-

ically, in Section 3.2, we inject reasoning traces generated by a stronger RLM (called an *injecting* model) into another RLM (called a *receiving* model), allowing the receiving model to generate final translations conditioned on the injected reasoning. Surprisingly, even when the injected reasoning traces are of high quality and accompanied by strong translations, we do not consistently observe an improvement in the receiving model's performance. This suggests increasing test-time compute or transferring generic reasoning traces is insufficient for improving MT, raising questions about the universality of current reasoning paradigms.

Based on these observations, we hypothesize that the limitation is not the absence of reasoning per se, but a misalignment between generic reasoning formats and the structural requirements of translation. Our observation is furthermore reinforced by the success of Chain-of-thoughts reasoning that improves MT quality (Wu et al., 2025b).

To test our hypothesis, in Section 4, we design a synthetic data generation pipeline that enforces a *structured reasoning* style for MT. Our proposed dynamic reasoning template explicitly incorporates iterative refinements using a multi-turn translation setup (Briakou et al., 2024) and a dynamic format that adapts to the difficulty of the given translation examples, enhancing the efficiency of reasoning length. We show that reasoning can be beneficial for MT when it is explicitly shaped around error revision, constraint satisfaction, and iterative refinement, rather than leaving it up to the model for generic reasoning.

Looking into the reason behind the effectiveness of the structured reasoning, in Section 6.1, and through controlled ablation studies with varying reasoning trace quality and target translation quality, we show that having higher quality reasoning traces (e.g., proposed structure reasoning) leads to performance gain on translation quality, even when paired with weaker final outputs. However, the quality of the final translation has a more positive role on the overall performance.

We summarize our contributions as follows:

- We present a comprehensive evaluation and analysis of reasoning large language models for machine translation. Beyond benchmarking, we conduct structural analysis showing that existing reasoning formats in RLMs, without revision, self-correction, or exploration, do not benefit MT. We also demonstrate that injecting reasoning traces in the same format but with higher-quality translations does not consistently improve RLM's performance in MT (Section 3).
- We propose a structured reasoning framework tailored to machine translation, a designed synthetic reasoning generation pipeline that enforces iterative revision and self-correction. Training on the structured reasoning traces leads to a significant improvement in translation performance (Sections 4 and 6).
- We disentangle the roles of reasoning trace and final translation quality through controlled ablation studies on RLMs, showing that both have a complementary effect. Structured reasoning improves the performance over linear reasoning even when the target translation quality is fixed, and the gain amplifies when stronger targets are provided (Section 6.1).

Together, these findings challenge the assumption that test-time scaling via longer generic reasoning traces will reliably improve performance across tasks, and instead point to the need for task-specific structural priors when generating synthetic reasoning data. More generally, they motivate viewing reasoning not as a single transferable skill, but as a family of task-dependent deliberation strategies that must be learned in a task-aware manner.

2 Related Work

Reasoning Language Models. Prompting language models to generate intermediate steps (commonly referred to as Chain-of-Thought (CoT) (Wei et al., 2022)) has yielded significant performance gains across complex, reasoning-demanding tasks like mathematics (Cobbe et al., 2021; Yu et al., 2025), logic reasoning (Sprague et al., 2025), and programming (Min et al., 2024; Wang et al., 2025d). In a similar direction, it has been shown that reasoning capabilities can also emerge through pure reinforcement learning (RL) training (DeepSeek Team, 2025; Mistral-AI, 2025). Beyond performance gains, recent studies have also looked into reasoning trajectories, investigating their structures and features, and validating the intermediate steps (Marjanovic et al., 2026; Liao et al., 2025; Liu et al., 2024; Zhang et al., 2024).

Reasoning for MT. Several studies have explored CoT prompting for machine translation, reporting remarkable gains on the performance (Nguyen and Xu, 2025; Wang et al., 2025b).

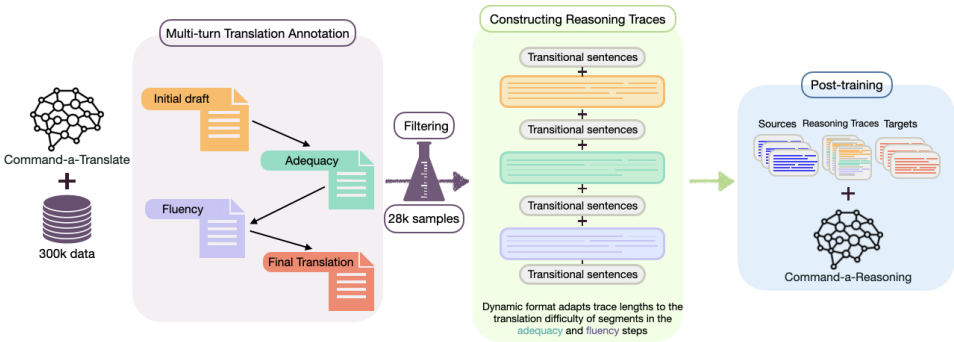

Figure 1: An overview of our structured reasoning framework tailored for machine translation.

Moreover, it has been shown that multi-step translation focusing on different aspects of translations in each step (e.g., drafting or fluency) (Briakou et al., 2024), as well as simple iterative translation refinements (Wu et al., 2025a), boost the translation quality. Designing reward modeling using MT metrics or human feedback in the RL-based methods is another effective way to improve the machine translation performance (He et al., 2024; Ramos et al., 2024, 2025; Feng et al., 2025).

In a nutshell, we show that thinking tokens are ineffective for MT across four reasoning model families, a conclusion that aligns with concurrent findings by Zebaze et al. (2025), whose benchmarking covers the Qwen3 family (Qwen Team, 2025) and DeepSeek-R1-Distill variants. Our broader evaluation across different reasoning families shows that the negative result is persistent. Beyond this, we take a further step and dig into the structural features of reasoning traces for MT. We characterize *how* current RLMs reason for MT (Section 3.1), showing traces are highly linear and lack exploration or self-correction. We also design a reasoning-injection experiment (Section 3.2) that demonstrates the observed failure to the format of reasoning rather than to the quality of translations in the traces. Building on these findings, we propose a structured reasoning framework based on adaptive successive self-revision over difficult segments of a given input, with explicit adequacy- and fluency-oriented steps. To show the effectiveness of our structured reasoning framework, we post-train a reasoning model on the generated structured traces and observe a solid performance improvement across a variety of languages. We further analyze the trained model's behavior through an ablation study that disentangles the contributions of structured reasoning traces and target translation quality to performance.

3 Reasoning Capability Inspection

Reasoning skills have gained special attention in recently developed large language models. It has been shown that explicit reasoning, also known as *thinking tokens*, through decomposition, self-verification, and iterative refinement substantially improves LLMs' performance across complex tasks, such as mathematical problems. However, the extent to which the reasoning capabilities benefit open-ended generation tasks like machine translation (MT) remains less explored.

To investigate the role of reasoning on MT, we begin with evaluating four open and closed-weights reasoning language models, Command-A-Reasoning (Cohere Team, 2025), Claude-4-Opus (Anthropic, 2025), DeepSeek-R1 (DeepSeek Team, 2025), and Gemini-2.5-Flash (Gemini Team, 2025). We evaluate these models on nine language pairs from the WMT24++ benchmark and compare the quality of generated translation with and without reasoning using the XCOMET-XL metric (Guerreiro et al., 2024), a state-of-the-art metric for automatic evaluation (Freitag et al., 2024) and the original recommended WMT24++ prompt. Additional experimental details are provided in Appendix A.

Section 3 summarizes the results. At a high-level glance and looking into the average performance, we observe that all the reasoning models consistently have a significantly higher performance without reasoning. The pattern holds for all the language pairs as well, except for Farsi in Claude-4-Opus where we see that reasoning improves the translation quality. Our findings show that while it has been widely reported in previous studies that reasoning abilities help LLMs to improve their performance on a variety of tasks (Huang et al., 2025; Son et al., 2026), it has

an opposite effect on the translation performance.

To further investigate the characteristics of reasoning traces in MT, we conduct a series of qualitative and quantitative analyses, focusing on the open-weights models, in the following sections.

3.1 Structural Analysis

To better understand why having reasoning traces fails to improve MT performance, we analyze the structure of reasoning traces for the translation task. Appendix B shows an example of a generated reasoning trajectory by DeepSeek-R1.

Looking into the generated reasoning traces, we notice that, unlike other reasoning-demanding tasks (like math problems), where the RLM explores multiple scenarios, evaluates intermediate steps, and can backtrack (An et al., 2025), MT reasoning style exhibits a linear structure. The RLMs split the source text into segments, translate each segment in isolation without considering possible alternative translations, and proceed sequentially without revisiting earlier translations.

To put this in a numerical perspective, we calculate the number of distinct reasoning paths (i.e., exploring different possible translations) that an RLM explores in its generated reasoning traces for MT. Following previous studies (An et al., 2025; Cuesta-Ramirez et al., 2025), we use linguistic cues, *Wait* and *Alternatively* as a signal for a reasoning path and count their occurrences. It is worth mentioning that since we evaluate on *En-X* direction, the models always reason in English.

We report the average number of different reasoning paths within the whole reasoning trace in Section 3.1. The numerical results confirm our qualitative observation that MT reasoning traces are highly linear, lacking exploration or self-correction in the reasoning process. Among the selected models, Command-A-Reasoning has the highest number of reasoning paths in the reasoning traces on average, meaning that the model explores alternative translations for an average of 1.56 segments in the given text. It is worth mentioning that a given text could have multiple segments.

3.2 Reasoning Injection

In our primary experiment, we have shown that reasoning does not improve performance on the machine translation task for RLMs, and this pattern is consistent across all the models. However, the absolute translation performance varies across models, with Claude-4-Opus achieving the highest overall performance. In this part, we investigate whether a higher-quality reasoning trace generated by a stronger model on MT provides other models with a hint to improve their performance.

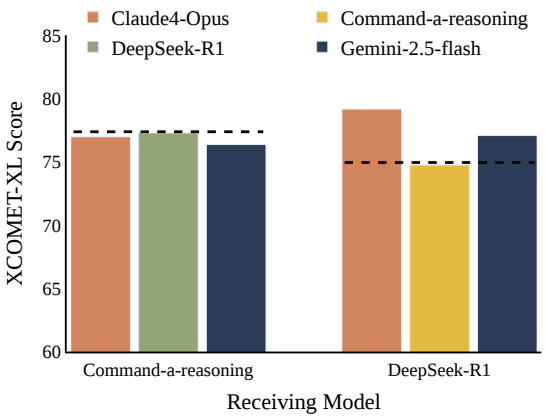

Figure 2: The performance of RLMs when their reasoning traces are generated and injected from other models. The dash-line represents the baseline performance of the receiving RLM with thinking tokens. The injecting RLMs are presented at the top. The strength of the injecting model used to generate reasoning traces does not correlate with final translation quality.

We conduct a controlled reasoning injection experiment. For each example, we generate a reasoning trace using an injecting model (e.g., Claude-4-Opus), insert it into a receiving model (e.g., Command-A-Reasoning), and let the receiving model generate the final translation using the provided reasoning trace. To have control over the reasoning traces between *thinking tokens*, we limit ourselves to the open-weights models, namely Command-A-Reasoning and DeepSeek-R1, as the receiving models, while using all four models as injecting ones to generate traces.

First, in Figure 2, we do not observe a notable performance drop in the receiving models, indicating that RLMs are not biased toward their own reasoning style for machine translation. Secondly, although reasoning traces from stronger injecting models, such as Claude-4-Opus, lead to performance improvement of DeepSeek-R1, this pattern does not hold for Command-A-Reasoning, meaning that reasoning traces generated by stronger models do not consistently yield higher translation quality across receiving models.

In summary, we find the reasoning styles for machine translation differ from other reasoning-demanding tasks. In their current format, translation reasoning trajectories follow a linear process and rarely include translation revision or ex-

Model	en-ar	en-cs	en-fa	en-fr	en-hi	en-ja	en-ko	en-ru	en-zh	Avg.
Command-A-Reasoning (w)	70.4	83.0	78.6	82.5	59.8	80.6	81.4	82.0	77.2	77.3
Command-A-Reasoning (w/o)	72.6	83.5	81.2	83.0	60.4	80.7	81.7	82.6	78.6	**78.3**
Claude-4-Opus (w)	74.9	84.9	81.5	83.2	64.4	83.9	83.5	84.2	80.7	80.1
Claude-4-Opus (w/o)	75.1	85.2	81.0	83.4	64.6	84.0	83.8	85.1	81.6	**80.4**
DeepSeek-R1 (w)	69.5	82.5	76.8	82.0	61.0	74.8	74.2	83.0	70.4	74.9
DeepSeek-R1 (w/o)	73.0	83.7	79.2	82.7	63.1	81.3	81.8	83.8	79.3	**78.7**
Gemini-2.5-Flash (w)	71.2	83.9	79.9	82.5	63.6	81.9	81.3	83.4	79.4	78.6
Gemini-2.5-Flash (w/o)	74.0	85.5	81.2	83.8	63.8	82.7	83.6	84.5	80.1	**79.9**

Table 1: Performance evaluation of reasoning language models on the WMT24++ benchmark using XCOMET-XL score. We observe that having reasoning traces hurts the translation quality across almost all languages and models.

Model	Avg. # Reasoning Paths	Std.
Command-A-Reasoning	**1.56**	0.23
Claude-4-Opus	0.006	0.0
DeepSeek-R1	0.107	0.16
Gemini-2.5-Flash	0.012	0.0

Table 2: Average number of reasoning paths in a generated reasoning trace per model, aggregated over the selected WMT24++ language pairs. All the RLMs show a high degree of linearity in their reasoning process.

ploration of possible translation options, and even higher-quality reasoning traces from stronger models cannot benefit other RLMs to enhance their translation, suggesting the ineffectiveness of the existing reasoning format in the MT task.

Building on our findings and inspired by previous studies on the effectiveness of step-wise translation methods, in the next section, we propose a framework for creating dynamic, step-by-step reasoning traces for machine translation.

4 Structured Reasoning Trace for MT

We design a synthetic data generation pipeline to train reasoning skills beneficial for machine translation in reasoning language models. Building on our findings in the previous section showing that the state-of-the-art reasoning language models do not generate useful reasoning traces for machine translation and the success of chain-of-thoughts techniques used in MT (Nguyen and Xu, 2025; Wu et al., 2025b; Briakou et al., 2024), we aim to create structured reasoning traces, teaching the model how to do step-by-step reasoning, and improve translation through successive refinements. In the following subsections, we describe our data generation process, quality check of reasoning traces, and the construction of reasoning templates.

4.1 Multi-turn Translation Annotation

We begin with a high-quality monolingual corpus of documents spanning 23 languages. All the

Dynamic Structured Reasoning Template

Step 1 – Initial Draft
Let me start by producing an initial, faithful translation of the source text into {target_language}, which will serve as the foundation for later refinements.
{initial_draft}

Step 2 – Adequacy
In this step, I focus on adequacy, ensuring that the full meaning of the challenging sentences is conveyed accurately and completely in {target_language}.
{challenging_sentences_adequacy}

Step 3 – Fluency
At this stage, I refine the challenging sentences, ensuring they read smoothly and naturally in {target_language}, as if originally written in it.
{challenging_sentences_fluency}

Step 4 – Final Translation
Finally, I bring together all the work from the previous steps, addressing any remaining issues or translation errors to produce a polished final version in {target_language}.

Figure 3: An example of the dynamic template used to generate structured reasoning traces. Curly-braced fields denote placeholders. Steps 2 and 3 are applied only to selected challenging sentences,; all other segments are discarded from the initial draft.

documents pass quality control and difficulty sampling (Proietti et al., 2025; Kocmi et al., 2025). Following Briakou et al. (2024), for each source document, we generate a sequence of four translation refinements using an offline, step-wise procedure employing the Command-A-Translate (Kocmi et al., 2025). Each step focuses on a distinct and important aspect of translation.

1. **Initial Draft:** Command-A-Translate is prompted to generate an initial draft intended to be rough and as a foundation for the next steps of translation.
2. **Adequacy Enhancement:** Given the draft translation, the model is asked to revise the translation with an explicit emphasis on semantic accuracy.
3. **Fluency Refinement:** In this step, the LLM improves the readability and fluency of the adequate translation in the target language.

4. **Final Translation:** In this final stage, Command-A-Translate considers all the previous translations, resolving remaining errors, and generates its best possible translation.

We consider this four-step trajectory as a structured approximation of a useful *reasoning process* for translation. In this step, we discard samples whose translated documents are not line-aligned with the source documents, a requirement for the follow up quality estimation, resulting in $320k$ high-quality samples. Detailed prompts used for each step of translations are presented in Appendix C.

4.2 Quality-based Data Selection

Having the multi-turn translations for $320k$ examples, we notice only a small subset shows meaningful improvements across the refinement steps. We quantify the improvement for each intermediate translation using the Quality Estimation (QE) version of MetricX (Juraska et al., 2024). Since MetricX has a limited context window, we evaluate all lines in isolation, averaging them into the document-level score.

We observe that less than 10% of examples show substantial quality gains (e.g., more than 0.5 MetricX improvement, technically meaning half a minor error in the MQM terminology (Lommel et al., 2013)) across different steps. This is expected as SOTA MT systems like Command-A-Translate have remarkably high performance and often translate well in a single pass, meaning most of the translation examples are not challenging for them. Therefore, we calculate the MetricX for each segment of a given document and consider the average MetricX score as the whole document score. As stated before, we keep only examples where the improvement between the initial draft and the final translation is at least 0.5 MetricX score. We find the filtering step critical for improving RLMs performance. This filtering yields a collection of 28k high-quality samples, each demonstrating a non-trivial translation. A breakdown of language-pair distribution for this subset is provided in Figure 4.

4.3 Constructing Reasoning Traces

Having instances where step-by-step translation is beneficial, we next convert the multi-step outputs into reasoning traces. We hypothesize that training the model with a step-by-step style of reasoning helps it use this structure for reasoning in the machine translation task. We later verify our hypothesis through empirical experiments.

Static Template The first way to create the reasoning trajectories is to chain the four translations into a single trace. We treat a set of human-generated sentences as connectives to connect four translations. To diversify the reasoning structure, for each step translation, we randomly insert one of the predefined sentences into the reasoning trace reflecting a *natural thinking flow* in the model. An illustration of this template is provided in Appendix D. A concern raised by this template is that the trace length scales with document size, which may introduce inefficiencies for long inputs.

Dynamic Template As mentioned above, having all four steps for the whole document is not efficient, as not all the segments of a document need translation through iterative reasoning. To enhance the efficiency of the reasoning traces, we develop a dynamic template that, after the initial draft, includes reasoning only for *challenging* segments. We consider sentences challenging that the translation quality between the initial drafting step and the adequacy or fluency stage improves by at least 1 MetricX point. More specifically, in this template, we include the initial draft translation of the entire document. Then, we drop the translation of segments without meaningful improvement in the fluency and adequacy steps from the reasoning trace. We also discard the final translation from the trace to reduce repetition. Figure 3 shows an example of the dynamic template strategy. More details on dynamic templates are presented in Appendix E.

4.4 Final Training Examples

Each training sample in our dataset consists of the triple *(source text, reasoning trace, target text)* in which reasoning traces are constructed using the explained dynamic template, and the target text is adopted as the Command-A-Translate final-step translation. The resulting 28k data provides explicit, structured reasoning for translation, which we use to train reasoning models to carry out step-by-step reasoning and improve their reasoning capability for machine translation.

5 Experimental Setups

As a post-training stage, we do supervised fine-tuning (SFT) on the Command-A-Reasoning model with 111B parameters using our curated 28k translation-reasoning examples. We pick this model as it is among the strongest open-weights reasoning models. All the experiments are con-

Setup	en-ar	en-cs	en-fa	en-fr	en-hi	en-ja	en-ko	en-ru	en-zh	Avg.
Baselines										
Base model	70.4	83.0	78.6	82.5	59.8	80.6	81.4	82.0	77.2	77.3
Direct Translation	76.7	86.3	82.7	85.7	66.2	83.7	84.5	85.5	81.9	81.5
Structured Reasoning $_{ours}$	77.8	87.1	83.6	86.4	66.3	83.8	84.7	86.2	82.1	**82.0**
Injected Reasoning										
Command-A-Reasoning	76.9	86.7	82.9	85.7	66.1	83.4	84.5	85.8	82.3	81.6
DeepSeek-R1	76.5	86.5	83.1	85.7	66.1	83.7	84.5	85.6	82.3	81.6
Structured Reasoning										
1k Samples	74.3	80.2	81.0	83.5	60.9	80.0	80.6	83.8	79.4	78.2
10k Samples	77.6	86.7	83.4	85.9	65.9	83.6	84.7	86.1	81.8	81.7

Table 3: The evaluation of reasoning-based translation systems. At the top, we have the Command-A-Reasoning as the backbone model and Direct Translations, where the base model is trained on MT without reasoning. The structured reasoning is the base model trained on our proposed structured reasoning. Injected Reasoning represents a comparison between the performance of the base model trained on injected reasoning from different reasoning models. At the bottom, a comparison between different training sizes of structured reasoning.

ducted with a maximum sequence length of 32768 tokens, and we evaluate the impact of different data selection strategies by training models on subsets of $1k$, $10k$, and the full $28k$ dataset. More details are presented in Appendix A.

5.1 Baselines

We compare our approach to following baselines:

Direct Translation To disentangle the effect of additional MT supervision from explicit reasoning, we finetune the same base model using only direct translations without reasoning traces. This setup is a strong baseline, as it benefits from the same high-quality Command-A-Translate target translations. We employ a set of ten templates within the thinking tokens, instructing the model to skip step-by-step reasoning. More details in Appendix F.

Injected Reasoning Traces As a contrastive ablation, we train the same base model on reasoning traces generated for the same 28k examples using Command-A-Reasoning and DeepSeek-R1. Note that only the reasoning traces are generated from these models, and the *target text* translation is kept from the Command-A-Translate as in the other experiments. This allows us to assess the role of structured reasoning traces compared to linear reasoning trajectories.

6 Results

We evaluate all models on the WMT24++ benchmark (Deutsch et al., 2025), focusing on nine diverse language pairs. Translation quality is evaluated using the XCOMET-XL metric (Guerreiro et al., 2024). For all evaluations, we set the decoding temperature to 0 to minimize variance and maximize reproducibility.

Table 3 summarizes the results. In the first block of the table, we compare the performance of our structured reasoning approach against both the base model and the direct translation finetuning setup. Direct Translation is a strong baseline as we ensure the performance improvement is not solely due to additional high-quality training data. Across all language pairs, structured reasoning shows substantial performance improvements over these baselines, indicating that the structured dynamic template is effective in developing a reasoning process that meaningfully supports translation quality. It is worth mentioning that 0.5 XCOMET-XL score improvement is considered noticeable by humans as +1 BLEU improvement (Kocmi et al., 2024).

The middle part of the table compares our synthetic reasoning traces with the models trained on the trajectories generated by DeepSeek-R1 and Command-A-Reasoning. Comparing these models against the direct translation baseline, we see that the injected reasoning does not offer additional benefits. This finding is consistent with our earlier analysis in Section 3, where we show that the existing reasoning capabilities of reasoning models are not beneficial for the machine translation task.

In the bottom of Table 3, we conduct an ablation on the size of the training data. We see a monotonically increasing trend in the performance as we use more data for training, aligning with Li et al. (2025) observations on the positive effect of training size.

6.1 Reasoning vs. Translation Quality

To further examine the relative contribution of reasoning trace quality and target translation quality, we design a controlled ablation study using our $28k$ samples. As we mentioned earlier, each training example includes a triple of *(source text, reasoning*

Setup	en-ar	en-cs	en-fa	en-fr	en-hi	en-ja	en-ko	en-ru	en-zh	Avg.
R-Src: Command-A-Reasoning										
Command-A-Reasoning	73.4	83.4	79.5	83.4	60.3	80.4	81.0	82.6	78.6	78.1
Command-A-Translate	76.9	86.7	82.9	85.7	66.1	83.4	84.5	85.8	82.3	81.6
R-Src: DeepSeek-R1										
DeepSeek-R1	72.0	82.3	77.9	81.9	59.9	80.2	81.0	81.9	76.9	77.1
Command-A-Translate	76.5	86.5	83.1	85.7	66.1	83.7	84.5	85.6	82.3	81.6
R-Src: Structured reasoning										
DeepSeek-R1	74.3	84.0	80.4	83.1	62.2	82.1	82.6	83.3	76.4	78.7
Command-A-Reasoning	74.4	84.3	81.0	83.4	61.6	81.9	82.4	83.3	80.0	79.1

Table 4: Comparing the role of reasoning and final translation quality on the performance. *R-Src* represents the injecting model of reasoning traces. Each row represents the model that generated the final translations. Both the quality of reasoning traces and final translations contribute positively to performance across languages, indicating that reasoning traces from stronger models improve performance and that higher-quality final translations in the training data lead to better performance.

trace, target text). Therefore, the quality of both the reasoning trace and the target text affects the model performance.

In this analysis, we vary these two factors independently. We consider three sources of reasoning traces: reasoning traces generated by Command-A-Reasoning, DeepSeek-R1, and our proposed structured reasoning. Separately, we consider three sources of target translations: translations generated by Command-A-Reasoning, DeepSeek-R1, and the stronger model Command-A-Translate. By pairing each reasoning trace source with each target translation source, we build multiple training datasets in which the reasoning traces and target translations may originate from different models. We then run SFT on the same base model using each dataset and evaluate the resulting model's performance.

The results are reported in Table 4. Each block is dedicated to a reasoning trace source, and each row within a block presents the target translation source model. Focusing on the top two blocks, we observe that for a fixed reasoning source, using target translations generated by the stronger model (i.e., Command-A-Translate) leads to higher performance than using target translations from the other models. Interestingly, for the Command-A-Translate's target translation set, we do not observe meaningful performance differences between reasoning traces generated by Command-A-Reasoning and DeepSeek-R1.

In the bottom block of Table 4, we evaluate structured reasoning paired with target translations from Command-A-Reasoning. This configuration yields higher performance than using DeepSeek-R1 translations. However, the gains remain notably smaller than those obtained when using Command-A-Translate as the target translation (Table 3). Interestingly, when target quality is fixed, structured reasoning improves over linear reasoning by about 1.3 XCOMET-XL on average, disentangling the contribution of reasoning format from translation quality.

Overall, these results indicate that reasoning format and target translation quality contribute complementarily to downstream performance. Structured reasoning consistently improves performance, and the gain amplifies when accompanied by strong target translations. This suggests that effective reasoning-based MT requires both task-structured traces and high-quality target translations.

7 Conclusions

In this paper, we conduct an in-depth study on the role of reasoning capabilities of reasoning language models in machine translation. Despite the substantial benefits that reasoning has provided for most reasoning-intensive tasks, our experiments show that current reasoning degrades translation quality across RLMs and languages. Through detailed analyses, we notice that the structure of current reasoning traces is highly linear, lacking key features such as revision, self-correction, and exploration of possible alternatives. Building on our findings, we propose a structured reasoning framework specifically designed for MT that dynamically adapts the length of reasoning to the difficulty of translation. We show that post-training an RLM on a small set of structured reasoning substantially improves the translation performance. Finally, we disentangle the contributions of reasoning trace quality and target translation quality through controlled ablations. We find that the two factors complement each other. Structured reasoning provides consistent gains, while higher-quality target translations amplify the gains further.

Impact Statement

In this work, we present in-depth analyses on the reasoning capabilities of RLMs for MT. We propose a structured reasoning generation pipeline to enhance translation performance in RLMs. We do not foresee any significant negative societal impacts directly resulting from this work.

Limitations

In this work, we use WMT24++ as our primary evaluation benchmark, focusing on 9 language pairs. While we have tried to select a diverse set of language pairs, adding more could strengthen our conclusions and findings. Moreover, we solely employ automatic metrics for evaluation and to investigate the role of reasoning on the translation quality. While costly, incorporating human evaluation could provide additional support for our findings.

References

Sohyun An, Ruochen Wang, Tianyi Zhou, and Cho-Jui Hsieh. 2025. Don't think longer, think wisely: Optimizing thinking dynamics for large reasoning models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Anthropic. 2025. System Card: Claude Opus 4 & Claude Sonnet 4. Technical report, Anthropic.

Eleftheria Briakou, Jiaming Luo, Colin Cherry, and Markus Freitag. 2024. Translating step-by-step: Decomposing the translation process for improved translation quality of long-form texts. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1301–1317, Miami, Florida, USA. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Cohere Team. 2025. Command a: An enterprise-ready large language model. *Preprint*, arXiv:2504.00698.

Jhouben Cuesta-Ramirez, Samuel Beaussant, and Mehdi Mounsif. 2025. LRMs are not thinking straight: Unreliability of thinking trajectories. In *Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long and Short Papers*, pages 27–32, Hannover, Germany. HsH Applied Academics.

DeepSeek Team. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Daniel Deutsch, Eleftheria Briakou, Isaac Rayburn Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. WMT24++: Expanding the language coverage of WMT24 to 55 languages & dialects. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12257–12284, Vienna, Austria. Association for Computational Linguistics.

Zhaopeng Feng, Shaosheng Cao, Jiahan Ren, Jiayuan Su, Ruizhe Chen, Yan Zhang, Jian Wu, and Zuozhu Liu. 2025. MT-r1-zero: Advancing LLM-based machine translation via r1-zero-like reinforcement learning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 18685–18702, Suzhou, China. Association for Computational Linguistics.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Marianna Buchicchio, Chrysoula Zerva, and Alon Lavie. 2024. Are LLMs breaking MT metrics? results of the WMT24 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 47–81, Miami, Florida, USA. Association for Computational Linguistics.

Gemini Team. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. xCOMET: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Zhiwei He, Xing Wang, Wenxiang Jiao, Zhuosheng Zhang, Rui Wang, Shuming Shi, and Zhaopeng Tu. 2024. Improving machine translation with human feedback: An exploration of quality estimation as a reward model. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8164–8180, Mexico City, Mexico. Association for Computational Linguistics.

Shulin Huang, Yiran Ding, Junshu Pan, and Yue Zhang. 2025. Beyond english-centric training: How reinforcement learning improves cross-lingual reasoning in llms. *Preprint*, arXiv:2509.23657.

Jin Jiang, Jianing Wang, Yuchen Yan, Yang Liu, Jianhua Zhu, Mengdi Zhang, and Liangcai Gao. 2025. Do large language models excel in complex logical reasoning with formal language? In *Proceedings of the 2025 Conference on Empirical Methods in Natural*

Language Processing, pages 16889–16914, Suzhou, China. Association for Computational Linguistics.

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

Kimi Team. 2025. Kimi k2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Tom Kocmi, Arkady Arkhangorodsky, Alexandre Berard, Phil Blunsom, Samuel Cahyawijaya, Théo Dehaze, Marzieh Fadaee, Nicholas Frosst, Matthias Galle, Aidan Gomez, Nithya Govindarajan, Wei-Yin Ko, Julia Kreutzer, Kelly Marchisio, Ahmet Üstün, Sebastian Vincent, and Ivan Zhang. 2025. Command-a-translate: Raising the bar of machine translation with difficulty filtering. In *Proceedings of the Tenth Conference on Machine Translation*, pages 789–799, Suzhou, China. Association for Computational Linguistics.

Tom Kocmi, Vilém Zouhar, Christian Federmann, and Matt Post. 2024. Navigating the metrics maze: Reconciling score magnitudes and accuracies. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1999–2014, Bangkok, Thailand. Association for Computational Linguistics.

Yang Li, Youssef Emad, Karthik Padthe, Jack Lanchantin, Weizhe Yuan, Thao Nguyen, Jason Weston, Shang-Wen Li, Dong Wang, Ilia Kulikov, and Xian Li. 2025. Naturalthoughts: Selecting and distilling reasoning traces for general reasoning tasks. *Preprint*, arXiv:2507.01921.

Zihao Li, Shaoxiong Ji, and Jörg Tiedemann. 2026. Test-time scaling of reasoning models for machine translation. *Preprint*, arXiv:2510.06471.

Baohao Liao, Xinyi Chen, Sara Rajaee, Yuhui Xu, Christian Herold, Anders Søgaard, Maarten de Rijke, and Christof Monz. 2025. Lost at the beginning of reasoning. *Preprint*, arXiv:2506.22058.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Sinuo Liu, Chenyang Lyu, Minghao Wu, Longyue Wang, Weihua Luo, Kaifu Zhang, and Zifu Shang. 2025. New trends for modern machine translation with large reasoning models. *Preprint*, arXiv:2503.10351.

Arle Richard Lommel, Aljoscha Burchardt, and Hans Uszkoreit. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, London, UK. Aslib.

Sara Vera Marjanovic, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, Benno Krojer, Xing Han Lù, Nicholas Meade, Dongchan Shin, Amirhossein Kazemnejad, Gaurav Kamath, Marius Mosbach, Karolina Stanczak, and Siva Reddy. 2026. Deepseek-r1 thoughtology: Let's think about LLM reasoning. *Transactions on Machine Learning Research*.

Marcus J. Min, Yangruibo Ding, Luca Buratti, Saurabh Pujar, Gail Kaiser, Suman Jana, and Baishakhi Ray. 2024. Beyond accuracy: Evaluating self-consistency of code large language models with identitychain. In *The Twelfth International Conference on Learning Representations*.

Mistral-AI. 2025. Magistral. *Preprint*, arXiv:2506.10910.

Lam Nguyen and Yang Xu. 2025. Reasoning for translation: Comparative analysis of chain-of-thought and tree-of-thought prompting for LLM translation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 259–275, Vienna, Austria. Association for Computational Linguistics.

NVIDIA. 2025. Nvidia nemotron nano 2: An accurate and efficient hybrid mamba-transformer reasoning model. *Preprint*, arXiv:2508.14444.

Lorenzo Proietti, Stefano Perrella, Vilém Zouhar, Roberto Navigli, and Tom Kocmi. 2025. Estimating machine translation difficulty. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 24261–24285, Suzhou, China. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Miguel Ramos, Patrick Fernandes, António Farinhas, and Andre Martins. 2024. Aligning neural machine translation models: Human feedback in training and inference. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pages 258–274, Sheffield, UK. European Association for Machine Translation (EAMT).

Miguel Moura Ramos, Tomás Almeida, Daniel Vareta, Filipe Azevedo, Sweta Agrawal, Patrick Fernandes, and André F. T. Martins. 2025. Fine-grained reward optimization for machine translation using error severity mappings. *Preprint*, arXiv:2411.05986.

Guijin Son, Donghun Yang, Hitesh Laxmichand Patel, Amit Agarwal, Hyunwoo Ko, Chanuk Lim, Srikant Panda, Minhyuk Kim, Nikunj Drolia, Dasol Choi, Kyong-Ha Lee, and Youngjae Yu. 2026. Pushing on multilingual reasoning models with language-mixed chain-of-thought. *Preprint*, arXiv:2510.04230.

Zayne Rea Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann

Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2025. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. In *The Thirteenth International Conference on Learning Representations*.

Qiushi Sun, Zhangyue Yin, Xiang Li, Zhiyong Wu, Xipeng Qiu, and Lingpeng Kong. 2024. Corex: Pushing the boundaries of complex reasoning through multi-model collaboration. In *First Conference on Language Modeling*.

Fali Wang, Hui Liu, Zhenwei DAI, Jingying Zeng, Zhiwei Zhang, Zongyu Wu, Chen Luo, Zhen Li, Xianfeng Tang, Qi He, and Suhang Wang. 2025a. AgentTTS: Large language model agent for test-time compute-optimal scaling strategy in complex tasks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jiaan Wang, Fandong Meng, Yunlong Liang, and Jie Zhou. 2025b. DRT: Deep reasoning translation via long chain-of-thought. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 6770–6782, Vienna, Austria. Association for Computational Linguistics.

Yiming Wang, Pei Zhang, Siyuan Huang, Baosong Yang, Zhuosheng Zhang, Fei Huang, and Rui Wang. 2025c. Sampling-efficient test-time scaling: Self-estimating the best-of-n sampling in early decoding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Yunkun Wang, Yue Zhang, Zhen Qin, Chen Zhi, Binhua Li, Fei Huang, Yongbin Li, and Shuiguang Deng. 2025d. ExploraCoder: Advancing code generation for multiple unseen APIs via planning and chained exploration. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18124–18145, Vienna, Austria. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Di Wu, Seth Aycock, and Christof Monz. 2025a. Please translate again: Two simple experiments on whether human-like reasoning helps translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20435–20451, Suzhou, China. Association for Computational Linguistics.

Minghao Wu, Jiahao Xu, Yulin Yuan, Gholamreza Haffari, Longyue Wan, Weihua Luo, and Kaifu Zhang. 2025b. (perhaps) beyond human translation: Harnessing multi-agent collaboration for translating ultra-long literary texts. *Transactions of the Association for Computational Linguistics*, 13:901–922.

Yiyao Yu, Yuxiang Zhang, Dongdong Zhang, Xiao Liang, Hengyuan Zhang, Xingxing Zhang, Mahmoud Khademi, Hany Hassan Awadalla, Junjie Wang, Yujiu Yang, and Furu Wei. 2025. Chain-of-reasoning: Towards unified mathematical reasoning in large language models via a multi-paradigm perspective. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24914–24937, Vienna, Austria. Association for Computational Linguistics.

Armel Zebaze, Rachel Bawden, and Benoît Sagot. 2025. Llm reasoning for machine translation: Synthetic data generation over thinking tokens. *Preprint*, arXiv:2510.11919.

YiFan Zhang, Hanlin Zhang, Li Li, and Eric Xing. 2024. Evaluating step-by-step reasoning through symbolic verification. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2984–3002, Mexico City, Mexico. Association for Computational Linguistics.

A Experimental Setups

In this paper, we consider nine language pairs to study the reasoning capabilities of RLMs using WMT24++ benchmark, including Arabic (ar), Czech (cs), Farsi (fa), French (fr), Hindi (hi), Japanese (ja), Korean (ko), Russian (ru), and Chinese (zh). In selecting language pairs, we made sure to cover a diverse set of languages. The Models trained on $28k$ samples are finetuned for 4 epochs. Models trained on $1k$ and $10k$ subsets are trained for 8 and 6 epochs, respectively. We use a peak learning rate of 1×10^{-5}, decayed to 1×10^{-6}, with additive weight decay of 0.1. All the SFT runs are done on 64 first class GPUs. We set the temperature to 0 to enhance the reproducibility of our results.

A.1 Evaluation prompt

We employ the originally suggested WMT24++ evaluation prompt with a slight adjustment (Deutsch et al., 2025), presented in the following box.

WMT24++ Evaluation Prompt

You are a professional src_lang to tgt_lang translator, tasked with providing translations suitable for use in tgt_region (tgt_code). Your goal is to accurately convey the meaning and nuances of the original src_lang text while adhering to tgt_lang grammar, vocabulary, and cultural sensitivities. Produce only the tgt_lang translation, without any additional explanations or commentary. Please translate the following src_lang text into tgt_lang (tgt_code):

A.2 Training data distribution

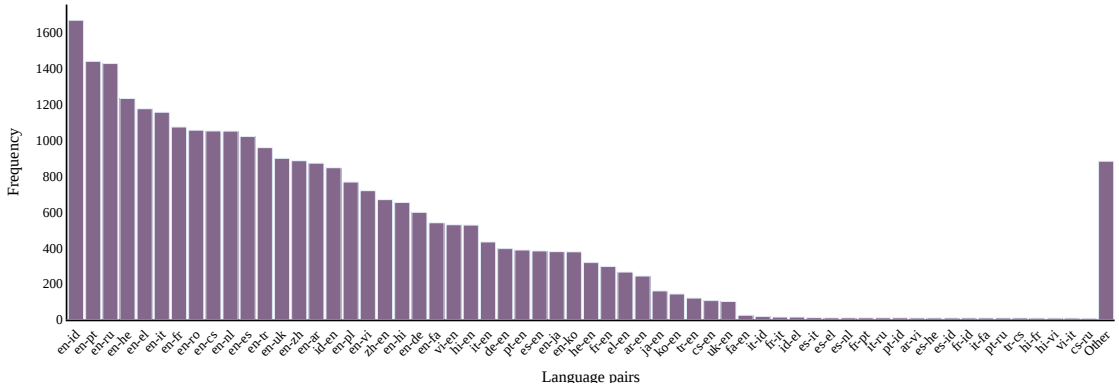

Figure 4: The distribution of language pairs in 28k structured reasoning traces as the training data. Most language pairs have a very low frequency. We put them in the *Other* category for presentation.

B A Reasoning Example

A generated reasoning trace and final translation for the given en-fr example by DeepSeek-R1.

Source Text (Input)

While the Louisiana Republican can avoid confronting the issue for now, on January 19 the country will face its deadline to pass four of 12 annual spending bills. Johnson has publicly stated that he would not support a "continuing resolution" (CR) to extend the deadline on these four bills to the February 2 due date for the remaining eight.

Reasoning

Hmm, the user wants me to translate an English political text about US budget deadlines into French. They've specified I should act as a professional translator focusing on French (France) conventions without any explanations.

This text discusses Louisiana Republican Mike Johnson's position on government spending bills. The key elements are: temporary avoidance of an issue, January 19 deadline for four bills, Johnson's rejection of a "continuing resolution" (CR), and the February 2 deadline for eight other bills.

For the translation, I need to consider several aspects. "Louisiana Republican" should become "le républicain de Louisiane" to maintain political context. "Confronting the issue" translates best to "confronter ce problème" in French political discourse. The term "continuing resolution" requires keeping the English abbreviation "CR" with French explanation in parentheses - "une 'continuing resolution' (CR)" - as this is standard practice for technical terms.

The dates need European formatting: "19 janvier" and "2 février". For "publicly stated", "a publiquement déclaré" sounds more natural than literal translations. The structure about extending deadlines to February 2 requires careful handling - "reporter l'échéance au 2 février" flows better than word-for-word translation.

I should ensure all verbs agree properly in tense and that the sentence structure maintains the original's urgency while sounding natural in French. The final translation needs to be polished and ready for professional use without any translator's notes.

Generated Translation (Final output)

Si le républicain de Louisiane peut éviter de confronter ce problème pour l'instant, le pays devra faire face, le 19 janvier, à l'échéance pour l'adoption de quatre des douze projets de loi de dépenses annuelles. Johnson a publiquement déclaré qu'il ne soutiendrait pas une "continuing resolution" (CR) pour reporter l'échéance de ces quatre textes au 2 février, date limite pour les huit autres.

C Translation Prompts

We use the following prompts to generate translations at each step of our 4-step template using Command-A-Translate.

Step-wise Translation Prompt Templates

Draft (Initial Translation)

```
Your goal is to translate a piece of text into {target_language} that accurately conveys the
meaning and structure of the source text. Your translation should closely reflect the original
content without omitting or adding information. If certain contextual details are missing, aim
for a general translation that remains flexible across different contexts, and do not include any
explanations or commentary. Keep the same number of lines as in the source text.
Provide your best single translation of the original text enclosed in triple backticks:

```{source_text}```
```

Refinement (Adequacy-Focused Translation)

```
Now, produce a new translation into {target_language} that focuses primarily on the adequacy of
translation.
```

Refinement (Fluency-Focused Translation)

```
Now, produce a new translation into {target_language} that focuses primarily on the fluency of
translation, ensuring it reads as if it were originally written in {target_language}. Provide
only one refined translation enclosed in triple backticks and do not output anything else after that.
```

Proofreading (Final Translation)

```
You are tasked with proofreading and final editing of a translation. You will be provided
with three different translations: first draft, translation focused on adequacy, and translation
focused on fluency. Your goal is to provide a polished final translation of the source text into
{target_language}. Keep the same number of lines as in the source text.
For your reference, below are the source text, the draft translation, and refined translations,
each enclosed in triple backticks:

Source ({source_language}):
```{source_text}```

Draft translation:
```{preprocess_analysis}```

Refined translation (adequacy):
```{draft_translation}```

Refined translation (fluency):
```{refined_translation}```

Proofread the refined text for grammar, spelling, punctuation, terminology, and overall fluency.
Ensure the translation accurately reflects the original meaning, intent, style, and structure,
while sounding fully natural and idiomatic in {target_language}. Provide only the final polished
translation enclosed in triple backticks and nothing else.
```
```

D Static Template

Here we bring an example of the static template. At each step, all the document translation is presented, making the reasoning trace lengthy and inefficient.

Static Reasoning Template

Step 1 — Initial Draft
Let me start by producing an initial, faithful translation of the source text into {target_language}, which will serve as the foundation for later refinements.
{initial_draft}

Step 2 — Adequacy
In this step, I focus on adequacy, ensuring that the full meaning of the challenging sentences is conveyed accurately and completely in {target_language}.
{adequacy_focused_translation}

Step 3 — Fluency
At this stage, I refine the challenging sentences, ensuring they read smoothly and naturally in {target_language}, as if originally written in it.
{fluency_focused_translation}

Step 4 — Final Translation
Finally, I bring together all the work from the previous steps, addressing any remaining issues or translation errors to produce a polished final version in {target_language}.
{final_translation}

E Structured Reasoning Templates

To make a diverse training set and avoid overfitting on the predefined sentences in the structured reasoning template, we consider a diverse set of transitional sentences for each step. We randomly pick a sentence for each step to generate a reasoning trace. Below, you find the set of sentences we considered.

Transitional Sentences for Structured Reasoning

Initial Draft

- Let me start by producing an initial, faithful translation of the source text into {target_language}, which will serve as the foundation for later refinements.
- To begin, I'll create an initial, precise translation of the given text into {target_language}, laying the groundwork for further adjustments.
- I'll translate the source text faithfully into {target_language} to form the base for refinement.
- For the first step, I'll produce a faithful initial translation into {target_language}.
- Let's begin with an accurate initial translation of the source into {target_language}, which I'll refine in later steps if needed.

Adequacy Refinement

- In this step, I focus on adequacy, ensuring that the full meaning of the challenging sentences is conveyed accurately and completely in {target_language}.
- My attention here is on adequacy, making sure the difficult sentences preserve every nuance and idea from the source in {target_language}.
- This step centers on adequacy, confirming that each complex sentence accurately communicates the original message in {target_language}.
- Adequacy guides my work in this step. I translate the challenging sentences so their sense remains complete and precise in {target_language}.

Fluency Refinement

- At this stage, I refine the challenging sentences, ensuring they read smoothly and naturally in {target_language}, as if originally written in it.
- This step focuses on improving fluency, making the difficult sentences sound idiomatic in {target_language}.
- I rework the challenging parts to produce fluent, native-like versions in {target_language}.
- I dedicate this step to refining the harder sentences, aiming for smooth and natural expression in {target_language}.

Final Translation

- Finally, I bring together all the work from the previous steps, addressing any remaining issues to produce a polished final version in {target_language}.
- I conclude by integrating all prior revisions and delivering a coherent, high-quality translation in {target_language}.
- This step pulls together all prior improvements, correcting residual errors and finalizing the translation in {target_language}.

F Direct Translation Baseline

Here we present the templates we use for direct translation baseline. In this setup, we have discarded the reasoning traces, and one of the randomly picked following sentences is put within the thinking tokens.

Direct Translation Templates (No Reasoning)

- The user is asking for a translation from {source_language} into {target_language}. Analyzing the source text, I noticed that it is not particularly challenging to translate. Therefore, I will skip the reasoning steps and proceed directly with the translation.
- This looks like an easy translation request. I'll skip step-by-step reasoning and directly translate it into {target_language}.
- Translate this from {source_language} to {target_language}. Since the text doesn't appear complex, I'll skip detailed reasoning and translate it directly.
- A translation from {source_language} to {target_language} is requested. The source seems easy, so I'll provide a direct translation right away.
- Let's translate this into {target_language}. The text isn't difficult, so I'll proceed without intermediate reasoning.
- I understand the task: translate the given text to {target_language}. It's straightforward, so I'll skip the reasoning process.
- This is a straightforward translation query. I'll respond with a direct translation into {target_language}, without a stepwise explanation.
- The user needs a translation from {source_language} into {target_language}. I'll create the final translation directly, as the text poses no challenge.
- I'll translate this from {source_language} into {target_language}. Because the content is easy, I won't follow a step-by-step translation process.
- The task is to translate from {source_language} to {target_language}. The text looks simple enough, so I'll go straight to the translation.