ReflectMT: Internalizing Reflection for Efficient and High-Quality Machine Translation

Kunquan Li[1,3 * †] **Yingxue Zhang**[2*] **Zhibin Lan**[1,3] **Fandong Meng**[2‡] **Jinsong Su**[1,3‡]
[1]School of Informatics, Xiamen University, China
[2]WeChat AI, Tencent Inc, China
[3]Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan (Xiamen University), Ministry of Culture and Tourism, China
{likunquan,lanzhibin}@stu.xmu.edu.cn, jssu@xmu.edu.cn
{yxuezhang,fandongmeng}@tencent.com

Abstract

Recent years have witnessed growing interest in applying Large Reasoning Models (LRMs) to Machine Translation (MT). While most approaches adopt a "*pre-thinking*" paradigm and benefit from explicit reasoning trajectories, they suffer from substantial inference cost and latency. To address these limitations, we propose ReflectMT, a two-stage reflection internalization framework for machine translation that employs a "*post-thinking*" paradigm. Our approach develops the model's "*translate–reflect–refine*" capability through reinforcement learning. In the first stage, we cultivate the model's capacity for high-quality reflection and refinement, thereby enhancing its semantic comprehension and task-specific knowledge. In the second stage, we train the model to internalize the knowledge acquired during reflection. On WMT24, ReflectMT achieves a GPT-4o-based GRF score of 86.16, outperforming DeepSeek-R1 (84.03) by 2.13 points, while using approximately 92% fewer output tokens. Independent automatic evaluation, a blinded human evaluation, and preliminary experiments on three additional translation directions provide further evidence for the framework's effectiveness.

1 Introduction

In recent years, Large Reasoning Models (LRMs), such as OpenAI-o1 [16] and DeepSeek-R1 [3], have demonstrated powerful capabilities in complex tasks such as mathematics, programming, and logical reasoning. These models typically leverage long Chain-of-Thought (CoT) processes, exploring different solution paths through reasoning, self-verification, and iterative refinement before generating the final answer.

Inspired by these advances, researchers have begun exploring the introduction of long CoT into Machine Translation (MT). We categorize these methods as the "*pre-thinking*" paradigm, wherein the model performs explicit reasoning before generating the translation. For instance, Marco-o1 [34] validated the effectiveness of long CoT in slang translation; Wang et al. [23] trained the DRT model through supervised fine-tuning for literary translation; However, although the explicit reasoning process significantly enhances the model's ability to handle difficult translation tasks, the "*pre-thinking*" paradigm requires the model to generate explicit and lengthy chains of thought during the inference stage, resulting in substantial computational overhead and latency. Although recent studies in general domains have explored efficient reasoning strategies—such as adaptive mode selection [32]

*Equal contribution.
†Work was done while Kunquan Li was an intern at WeChat AI, Tencent Inc.,China.
‡Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and dynamic reasoning depth control [33]—they merely regulate the length or trigger of reasoning and overlook the multi-dimensional thinking aspects (e.g., semantic fidelity, cultural adaptation) that are central to translation, leaving the efficiency–quality trade-off in MT unresolved.

To address the above limitations, this paper proposes a novel "*post-thinking*" algorithm for machine translation, named ReflectMT. Unlike the "*pre-thinking*" paradigm, our core motivation is to leverage the reflection mechanism to enhance model capabilities during training, enabling the direct generation of high-quality translations during inference without incurring additional computational costs. The framework is divided into two phases, in the first stage (reflection capability establishment), we guide the model to learn the complete "*translate-reflect-refine*" process through carefully designed reward functions. In the second stage (capability internalization and transfer), we guide the model to integrate the acquired reflection knowledge into the initial translation process by reinforcing the reward signal for direct translation.

Specifically, we adopt a "*translate-reflect-refine*" workflow during the training stage: the model first generates an initial translation, then initiates a structured reflection process to critically evaluate it, and finally generates a refined translation. Different from free-form reflection in general tasks, we design a multi-dimensional structured reflection process tailored to the specific needs of MT (covering error identification, ambiguity analysis, etc.) and construct a high-quality reflection dataset through multi-agent collaboration.

Furthermore, to achieve internalization of reflection capabilities, this algorithm mirrors the skill development of human translators: novices rely on explicit reflection and revision, whereas accumulated experience allows these processes to be internalized, enabling the direct generation of high-quality translations. This training paradigm achieves the transformation from "*explicit reflection*" to "*implicit capability*", enabling the model to directly output high-quality translations during inference.

Experimental results on multiple English-to-Chinese benchmarks show that ReflectMT substantially reduces inference-token consumption while maintaining competitive translation quality. On WMT24, ReflectMT achieves a GPT-4o-based GRF score of 86.16, compared with 84.03 for DeepSeek-R1, using 47.57 rather than 605.42 output tokens per sentence. Under the matched explicit-reflection setting, ReflectMT obtains a 0.24-point GRF advantage over Qwen3-8B with reflection (86.16 vs. 85.92), while reducing output tokens from 786.63 to 47.57 per sentence. Independent automatic evaluation, a blinded human evaluation, and preliminary experiments on three additional translation directions further support these findings.

The main contributions of this paper are summarized as follows:

- We introduce a "*post-thinking*" reflection mechanism into neural machine translation, designing a multi-dimensional structured reflection process tailored to MT tasks that enables systematic evaluation and iterative improvement of translations.
- We propose a reflection internalization training framework based on RL, which internalizes reflection knowledge into the model's direct generation capability, significantly improving translation quality while eliminating additional computational overhead during inference.
- We construct a high-quality MT reflection dataset via multi-agent collaboration, providing a robust data foundation for cultivating the model's reflection capability. Extensive ablation studies on the dataset validate the effectiveness of our proposed framework.

2 Data Construction

High-quality reflective data is crucial for equipping models with structured reflection and refinement capabilities. However, existing machine translation datasets typically only contain parallel corpora and lack intermediate reasoning steps. To address this issue, we designed an iterative multi-agent collaboration system that automatically constructs datasets containing complete reflection chains through dialogues between the Translator and the Reflector.

As illustrated in Figure 1, for each pre-collected sentence (denoted as x), we adopt a dual-agent framework for translation. The synthesis process is as follows:

(1)Initial Translation: The Translator generates an initial translation of the source sentence x, denoted as $y_0 = \text{Translator}(x)$.

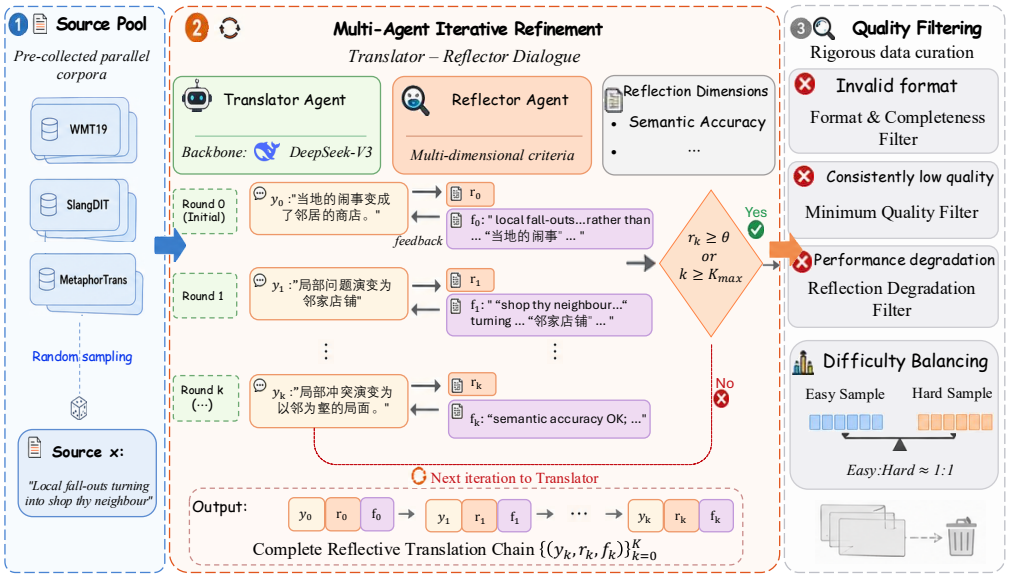


Figure 1: Overview of our reflective translation data construction. A Translator and a Reflector interact iteratively: the Translator produces a translation, the Reflector scores it and provides refinement suggestions, and this loop continues until the quality is satisfactory. The resulting data are then filtered to form the final reflective translation dataset.

(2)Reflective Evaluation: The Reflector critically evaluates y_0 to provide a multi-dimensional assessment. This includes a quality score $r_0 = \text{Reflector_score}(x, y_0) \in [0, 100]$ based on predefined criteria (such as semantic accuracy, cultural adaptability, and fluency), alongside structured refinement suggestions $f_0 = \text{Reflector_suggest}(x, y_0, r_0)$, that target specific lexical and structural deficiencies.

(3)Iterative Refinement Loop: The agents collaboratively optimize the translation through a feedback loop. At each iteration k ($k \geq 1$), the Translator generates an updated translation $y_k = \text{Translator}(x, y_{k-1}, f_{k-1}, r_{k-1})$. The Reflector then returns a new score r_k and updated suggestions f_k. This loop terminates when the score reaches a satisfactory threshold ($r_k \geq \theta$) or the maximum iteration limit is reached ($k \geq K_{\max}$). The sequence $\{(y_k, r_k, f_k)\}_{k=0}^{K}$ forms a complete reflective translation chain.

To ensure data quality, we apply rigorous filtering to exclude: (1) samples with invalid formats; (2) samples with consistently low scores ($r_K < r_{low}$); and (3) samples exhibiting performance degradation after reflection ($r_K < r_0$).

Hyperparameter settings are detailed in Appendix A.

3 Methodology

3.1 Overview

Building on the reflective translation data from Section 2, we present our *Reflection Internalization* framework, which turns an explicit "*translate–reflect–refine*" trajectory into a single-pass generation capability. As illustrated in Figure 2, the framework consists of (1) multi-dimensional reward modeling (Section 3.2); and (2) a two-stage RL framework that first *establishes* reflection (Section 3.3) and then *internalizes* it into the initial translation (Section 3.4).

3.2 Reward Modeling

To implement our RL algorithm, we design four reward components. Given a source sentence x, we require the model to generate outputs adhering to a strictly defined structural template:

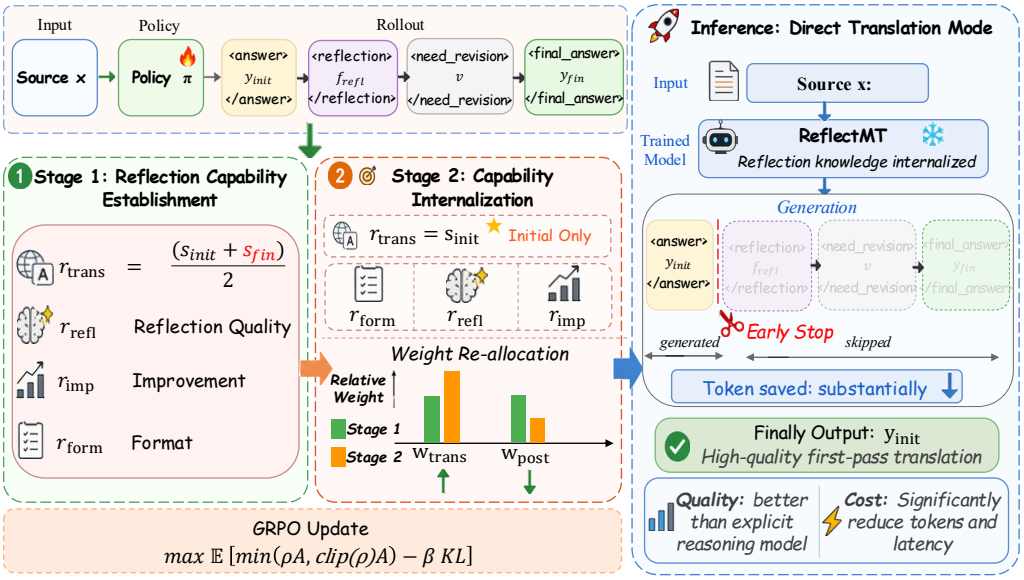


Figure 2: Training and inference paradigm of ReflectMT. During training, the model learns the complete "*translate–reflect–refine*" process in two stages: Stage 1 establishes the reflection capability, and Stage 2 internalizes it into the initial translation. During inference, an early-stopping strategy is applied so that the model directly outputs the first-pass translations without explicit reflection.

<answer>y_{init}</answer> <reflection>f_{refl} </reflection> <need_revision>v_{rev}</need_revision> <final_answer>y_{fin}</final_answer>.

We employ DeepSeek-V3 [10] as an LLM-as-a-Judge, denoted as $\mathcal{J}_{\text{v3}}(\cdot)$, to score texts on a 0-1 scale.

Format Reward (r_{form}). We use regular expressions to verify whether the output strictly contains all required XML tags in the correct order:

$$r_{\text{form}} = \begin{cases} 1 & \text{if format is correct} \\ 0 & \text{otherwise} \end{cases}. \tag{1}$$

Reflection Quality Reward (r_{refl}). We evaluate the reflection content f_{refl} based on the accuracy of problem identification and the actionability of the suggestions:

$$r_{\text{refl}} = \mathcal{J}_{\text{v3}}(x, y_{\text{init}}, f_{\text{refl}}). \tag{2}$$

Translation Quality Scores (s_{init} and s_{fin}). We score the quality (e.g., semantic accuracy, fluency) of both the initial and final translations: $s_{\text{init}} = \mathcal{J}_{\text{v3}}(x, y_{\text{init}})$ and $s_{\text{fin}} = \mathcal{J}_{\text{v3}}(x, y_{\text{fin}})$. To facilitate a smooth transition from explicit reflection to implicit internalization, the formulation of the Translation Quality Reward (r_{trans}) dynamically adapts according to the training stage (detailed in Sections 3.3 and 3.4).

Reflection Improvement Reward (r_{imp}). To encourage meaningful refinement, we define the score difference $\Delta s = s_{\text{fin}} - s_{\text{init}}$. The `<need_revision>` tag encodes a binary decision $v_{\text{rev}} \in \{0, 1\}$, where 1 indicates that the initial translation requires revision. We apply the improvement reward only when this decision is consistent with whether the final translation actually modifies the initial translation:

$$r_{\text{imp}} = \begin{cases} \Delta s, & \text{if } v_{\text{rev}} = \mathbf{1}[y_{\text{fin}} \neq y_{\text{init}}] \text{ and } \Delta s > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Total Reward (R). The total reward is a weighted sum of the aforementioned components:

$$R = r_{\text{form}} + w_{\text{trans}} r_{\text{trans}} + w_{\text{post}}(r_{\text{refl}} + r_{\text{imp}}), \tag{4}$$

where $w_{\text{trans}}, w_{\text{post}}$ denote the corresponding weights assigned to each reward component. (see Appendix G for detailed hyperparameter settings).

3.3 Stage 1: Reflection Capability Establishment

In Stage 1, our objective is for the model to master the complete reflective translation task. We use Qwen2.5-7B-Instruct [29] as the base model and perform a cold-start Supervised Fine-Tuning (SFT) via Low-Rank Adaptation (LoRA) [4], leveraging the data constructed in Section 2, enabling the model to generate the structured output format.

Subsequently, we apply the GRPO [19] algorithm. In this stage, we expect the model to produce a reasonable initial translation and a highly refined final translation. Therefore, the translation quality reward for Stage 1 is defined as the average of the initial and final translation scores:

$$r_{\text{trans}}^{\text{stage1}} = \frac{s_{\text{init}} + s_{\text{fin}}}{2} \,. \tag{5}$$

Given a source sentence x, GRPO samples n generations $\{g_1, \ldots, g_n\}$ from the policy π_{old}. We compute the Stage 1 total reward r_i for each g_i using Equation 4. GRPO optimizes the policy π_θ by maximizing:

$$\frac{1}{n}\sum_{i=1}^{n}\left[\min\Big(\rho_i A_i, \text{clip}(\rho_i, 1-\epsilon, 1+\epsilon)A_i\Big) - \beta\mathbb{D}_{\text{KL}}\big(\pi_\theta\big\|\pi_{\text{ref}}\big)\right], \tag{6}$$

where $\rho_i = \frac{\pi_\theta(g_i|x)}{\pi_{old}(g_i|x)}$, π_{ref} is the cold-start SFT model, and ϵ, β are hyperparameters. The advantage A_i is computed by normalizing the rewards:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_n\})}{\text{std}(\{r_1, r_2, \cdots, r_n\})}. \tag{7}$$

3.4 Stage 2: Capability Internalization and Transfer

In Stage 2, we shift the training focus to the quality of the initial translation. To force the model to internalize the reflection knowledge acquired in Stage 1 into its first-pass translations, we introduce two critical modifications to the reward mechanism.

First, the translation quality reward is solely determined by the initial translation score, disregarding the final translation:

$$r_{\text{trans}}^{\text{stage2}} = s_{\text{init}}. \tag{8}$$

Second, we adjust the weight configuration of the total reward by significantly increasing the translation weight $w_{\text{trans}}^{\text{new}}$ and correspondingly decreasing the improvement weight $w_{\text{post}}^{\text{new}}$. However, the model is still required to output the full structured tags during training to maintain RL stability and prevent catastrophic forgetting of the reasoning format. This reward restructuring incentivizes the model to maximize the quality of the first-pass translations y_{init}, thereby reducing its reliance on subsequent reflection steps.

3.5 Inference Phase

Owing to this internalization mechanism, the model directly produces high-quality first-pass translations. During inference, we externally terminate generation after the `</answer>` tag, retaining the complete translation while bypassing `<reflection>` and `<final_answer>` to reduce computational overhead and latency. This is an inference-time policy rather than learned termination: without truncation, the model continues generating these fields. Internalization thus refers to translation knowledge, not stopping behavior.

4 Experiments

4.1 Experimental Setups

Data. We constructed a reflective translation dataset specifically designed for English-to-Chinese machine translation, with the data volume shown in Table 11. Each sample comprises the source

sentence, the initial translation, the reflective analysis, and the final translation. In addition, we used official test sets from WMT23[4], WMT24[5], and FLORES-200 [21] to evaluate the model's generalization ability. To further probe generalization across translation directions, we additionally evaluated Chinese-to-English, English-to-Russian, and English-to-German translation, with results presented in Appendix J. Comprehensive dataset statistics and qualitative examples are provided in Appendix A.

Metrics. Following Wang et al. [23], we primarily evaluated the model's translation results using GRF (GPT Reference-Free), a GPT-based metric that does not require human references, with all GRF scores reported in this paper computed using GPT-4o. The evaluation prompt is detailed in Appendix D. Additionally, we reported two widely used semantic-level evaluation metrics based on pre-trained models: COMETKIWI-XL(CK) [18] and MetricX-24(MX24) [6]. In terms of efficiency, we evaluate using the number of tokens, calculated uniformly as "*average number of output tokens per sentence*".

Backbones. We use Qwen2.5-7B-Instruct as our base model. For detailed training information, please refer to Appendix G.

4.2 Baselines

To comprehensively evaluate the performance of our model, we selected three categories of baseline models, covering general large language models, large reasoning models, and specialized machine translation models.

General LLMs. We use Llama3.1-8B [2], Qwen2.5-7B, Qwen2.5-14B [29]. Additionally, we compare with the closed-source commercial model GPT-4o [16], with results presented in Appendix L.

LRMs. We use Marco-o1-7B [34], QwQ-32B [17], DeepSeek-R1 [3], and Qwen3-8B [30] as baselines.

Specialized MT models. Since existing efficient reasoning methods have not been explored in MT, we train two "*pre-thinking*" baselines based on Qwen2.5-7B-Instruct: an *Adapt* variant that learns to selectively engage in reasoning based on input complexity, and a *Full* variant (i.e., a full-thinking mode; see Appendix I). We also include DRT-7B [23], DeepTrans-7B [24], and ExTrans-7B [25], which follow a "*pre-thinking*" paradigm.

To ensure a rigorous and equitable comparison, we additionally evaluated the baseline models under a "*translate-reflect-refine*" setting, denoted as *w/ refl*. Specifically, during inference, we apply an identical structured reflective prompt to the baselines, and we strictly use the resulting refined translation for evaluation. Notably, both our method and the baselines demonstrate perfect compliance with this format, consistently outputting the initial translation, reflective analysis, and final translation without exception.

4.3 Main Result

Our ReflectMT models employ a direct translation mode without explicit reflection during inference. Experimental results show a favorable efficiency–quality trade-off across the evaluated benchmarks. To characterize this trade-off, we compare ReflectMT with four groups of baselines:

Comparison with general LLMs. As shown in Table 1, our optimal model ReflectMT-Stage2 achieves a GRF score of 82.55 on our dataset, representing a 12.3% improvement over the baseline model Qwen2.5-7B-Instruct (73.52). For MetricX-24 and CometKiwi metrics, our method achieves relative improvements of 26.0% and 2.1%, respectively. Notably, our model consumes an average of 30.73 tokens per sample during inference, nearly identical to the non-reflective general LLM (33.08 tokens/sample), introducing negligible additional inference overhead.

Comparison with strong LRM baselines. Our model demonstrates superior performance over strong LRMs, outperforming QwQ-32B (79.50) and DeepSeek-R1 (78.00) by 3.05 and 4.55 GRF points, respectively. To ensure a rigorous comparison, we also evaluated these baselines in a "*translate-reflect-refine*" mode. While this explicit reflection universally improves their translation

[4] https://www2.statmt.org/wmt23/translation-task.html
[5] https://www2.statmt.org/wmt24/translation-task.html

Table 1: Experimental results in English-to-Chinese translation. **Bold** and underlined values denote the best and second-best scores, respectively. Superscripts *, **, and *** indicate statistically significant improvements over DeepSeek-R1 with reflection under paired bootstrap tests at $p < 0.05$, $p < 0.01$, and $p < 0.001$, respectively. Detailed results are provided in Appendix C, Table 10.

Model	Our				WMT23				WMT24				FLORES-200			
	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
Qwen2.5-7B	73.52	2.70	76.43	33.08	75.54	2.65	78.83	28.33	69.51	3.23	76.51	39.37	77.82	2.37	82.09	25.69
Qwen2.5-14B	74.21	2.65	77.02	29.06	83.69	2.17	80.67	31.14	77.63	2.77	78.81	43.38	85.83	1.75	84.18	29.59
Qwen2.5-14B(w/ refl)	79.08	2.58	77.21	188.44	88.48	2.07	80.85	193.38	85.33	2.63	79.29	208.86	88.19	1.67	84.47	183.86
Llama3.1-8B	62.19	3.20	74.98	39.18	75.35	2.58	79.29	31.71	65.57	3.46	75.95	52.34	75.82	2.20	82.49	46.87
Qwen3-8B	79.54	2.53	77.90	628.63	88.20	2.00	81.18	588.59	82.56	2.59	79.58	654.06	89.49	1.61	84.64	612.31
Qwen3-8B(w/ refl)	81.78	2.42	78.03	863.12	89.00	2.00	**81.24**	738.92	85.92	2.51	79.66	786.63	90.78	1.67	84.72	726.28
Marco-o1-7B	68.43	2.90	75.80	489.24	79.58	2.38	79.42	548.34	74.67	2.97	77.35	584.30	82.72	1.86	83.50	535.60
Marco-o1-7B(w/ refl)	75.09	2.84	76.07	731.71	85.34	2.26	80.03	725.51	81.88	2.88	78.18	763.97	86.84	1.86	83.55	720.18
DeepSeek-R1	78.00	3.03	76.50	541.88	88.37	2.06	79.39	593.12	84.03	2.64	77.93	605.42	90.23	1.80	83.17	783.20
DeepSeek-R1(w/ refl)	81.15	2.50	76.67	659.83	**89.79**	1.93	81.04	1029.68	85.61	2.56	79.28	1069.07	90.50	1.56	**84.80**	982.33
QwQ-32B	79.50	2.22	77.77	523.72	85.51	2.79	78.59	632.56	78.90	3.03	74.82	684.29	89.03	2.19	82.84	547.19
QwQ-32B(w/ refl)	81.65	2.16	77.97	1063.01	87.17	2.60	78.33	1136.59	84.47	2.98	77.61	1102.00	89.86	2.11	82.94	1007.71
DeepTrans-7B	75.00	3.29	72.62	386.66	78.23	2.61	77.12	2018.96	76.22	2.99	77.39	2581.52	82.36	2.58	81.92	707.06
DRT-7B	75.69	2.93	75.48	478.99	79.70	2.54	78.87	479.64	76.39	3.00	77.40	520.03	85.15	2.22	82.85	463.02
ExTrans-7B	78.95	2.33	77.63	979.08	81.83	2.17	78.96	1547.55	81.70	2.72	77.51	1957.79	86.64	1.82	82.89	1232.43
Qwen-7B-Adapt-SFT	76.22	2.63	77.11	51.30	82.15	2.13	80.55	150.15	76.59	2.57	78.74	183.75	85.65	1.60	83.22	139.48
Qwen-7B-Adapt-RL	79.81	2.27	77.62	56.29	83.70	1.73	80.77	192.73	79.40	2.15	79.68	239.31	86.53	1.37	84.56	172.99
Qwen-7B-Full-RL	78.59	2.32	77.43	127.47	82.89	1.74	80.71	201.31	79.92	2.16	79.67	245.99	86.29	1.35	84.71	187.63
ReflectMT-SFT(Cold Start)	76.35	2.65	76.70	**23.25**	86.01	2.21	80.35	**25.94**	80.82	2.73	79.12	**37.57**	87.53	1.70	84.24	**24.09**
ReflectMT-Stage1(Our)	79.73	2.37	77.45	29.41	87.89	1.82	80.78	28.91	83.39	2.49	79.41	41.39	89.32	1.45	84.39	24.88
ReflectMT-Stage2(Our)	**82.55****	**2.00**	**78.05*****	30.73	89.48	**1.70**	81.18	30.67	**86.16***	**2.10**	**79.86*****	47.57	**91.42**	**1.33**	84.72	29.10

quality (e.g., DeepSeek-R1 w/ refl improves to 81.15), it necessitates generating lengthy chains of thought, resulting in substantially higher token consumption (e.g., DeepSeek-R1 consumes 541.88 tokens/sample, 17$\times$ that of our model). In contrast, our model, built upon the compact Qwen2.5-7B backbone, internalizes this reflection capability to generate high-quality translations directly, offering significant advantages in both parameter scale and computational efficiency.

Comparison with MT-specialized models. On our in-house test set, ReflectMT achieves a GRF score of 82.55, compared with 78.95 for ExTrans-7B, while generating substantially fewer output tokens. Results on other benchmarks and metrics demonstrate its appealing efficiency–quality trade-off.

Comparison with pre-thinking models. To verify the advantages of post-thinking reflection over pre-thinking, we trained a pre-thinking baseline under identical training configurations. Compared to the adaptive pre-thinking model (79.81), our method improves by 2.74 GRF points while reducing token consumption by 45%. On this evaluation setting, these results suggest that the post-thinking design can improve the efficiency–quality trade-off relative to the adaptive pre-thinking baseline.

We additionally conducted a blinded human evaluation on 150 examples from our in-house test set; the evaluation protocol and results are provided in Appendix B.

4.4 Human Evaluation

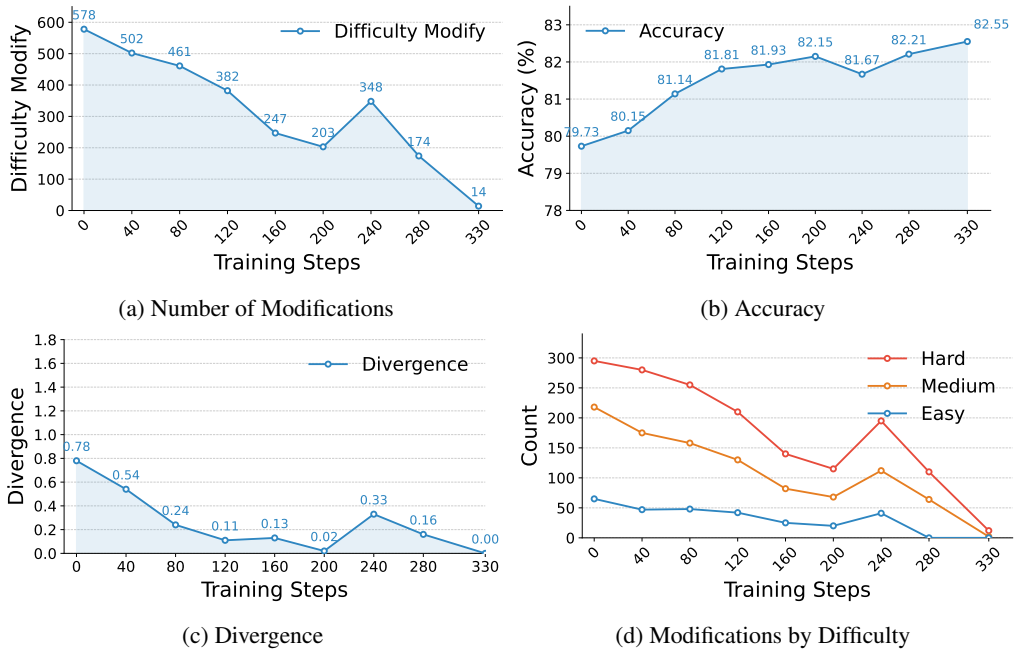
To complement automatic evaluation, we randomly sampled 150 source sentences from our in-house test set and compared ReflectMT with DeepSeek-R1 with reflection. Two bilingual NLP researchers independently scored the translations on a 0–100 scale, considering semantic faithfulness, fluency, and adequacy relative to the source. System identities were hidden and the presentation order was randomized.

Table 2: Blinded human evaluation on 150 in-house test sentences. Scores are averaged over two annotators (0–100; higher is better).

System	Human score
DeepSeek-R1 (w/ refl)	89.99
ReflectMT	**91.34**

As shown in Table 2, ReflectMT obtains an average score of 91.34, exceeding DeepSeek-R1 with reflection by 1.35 points. Both annotators assign higher average scores to ReflectMT. These results complement the automatic metrics, suggesting that our efficient first-pass translations retain high quality under human assessment on this sampled subset. Per-annotator scores and agreement analysis are provided in Appendix B.

4.5 Reflection Internalization Analysis



(a) Number of Modifications

(b) Accuracy

(c) Divergence

(d) Modifications by Difficulty

Figure 3: Training Dynamics: As the training steps increase (a) indicates the number of refinements; (b) represents the GRF score; (c) illustrates the score difference between initial and final translations; (d) shows the number of refinements for tasks of varying difficulty.

Training Dynamics of Reflection Internalization. We evaluate the model every 40 steps during Stage 2 on a 2000-sample test set, temporarily disabling early-stopping to observe both initial and final translations. As shown in Figure 3(b), GRF scores exhibit a fluctuating upward trend, reflecting the model's adjustment of internal representations to balance preliminary translation quality and reflective improvement ability during the gradual internalization of its reflective capability.

Table 3: Comparative performance of initial translations and final refinements across different training stages. The shrinking performance gap demonstrates the successful internalization of reflection capabilities.

Model	Our				WMT23				WMT24				FLORES-200			
	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
ReflectMT-SFT-initial	76.35	2.65	76.70	**23.25**	86.01	2.21	80.35	**25.94**	80.82	2.73	79.12	37.57	87.53	1.70	84.24	**24.09**
ReflectMT-SFT-final	77.26	2.63	77.32	297.51	86.05	2.16	80.87	273.78	83.15	2.69	79.42	323.75	87.83	1.69	84.30	248.34
ReflectMT-Stage1-initial	79.73	2.37	77.45	29.41	87.89	1.82	80.78	28.91	83.39	2.49	79.41	41.39	89.32	1.45	84.39	24.88
ReflectMT-Stage1-final	80.51	2.23	77.81	261.25	88.46	1.87	80.99	239.15	84.55	2.24	79.51	298.27	90.06	1.39	84.50	322.28
ReflectMT-Stage2-initial	**82.55**	**2.00**	**78.05**	30.73	89.48	**1.70**	**81.18**	30.67	86.16	**2.10**	79.86	47.57	91.42	**1.33**	**84.72**	29.10
ReflectMT-Stage2-final	82.55	2.00	78.05	322.29	**89.49**	1.70	81.18	288.35	**86.19**	2.10	**79.87**	350.77	**91.47**	1.33	84.72	244.50

Furthermore, we monitor the frequency of explicit modifications. Crucially, the model learns to self-assess, triggering refinement only when it judges the initial translation as sub-optimal. As shown in Figure 3(a), modifications drop monotonically from 578 (28.9%) at Step 0 to a mere 14 (0.7%) by Step 330. This trend indicates that as the reflective capability is internalized, the model generates higher-quality translations in the first pass. Figure 3(c) shows that the score difference between first-pass and final translations exhibits a fluctuating downward trend. As the reflective capability is gradually internalized, the model ultimately generates optimal translations in the first pass without subsequent modifications. Table 3 displays first-pass and final translation results at different training stages. This demonstrates the effectiveness of our reflective internalization training framework: the model successfully incorporates the knowledge learned during reflection into its initial translation capability, achieving quality close to the final translation during the first generation.

Performance Across Difficulty Levels. To evaluate model performance on translation tasks of varying difficulty, we divided samples into three levels based on GRF scores: easy (>90), medium

(70-90), and difficult (<70), with proportions approximately 11:6:3. We recorded the number of samples requiring reflective modifications during training at each difficulty level.

As shown in Figure 3(d), at Step 0, the model modified 65 easy, 218 medium, and 295 difficult samples, indicating significant room for improvement in the initial translation stage despite acquired reflective capability after first-phase training, especially for difficult samples.

After second-phase reflective internalization training, model performance at Step 330 showed qualitative improvement. For easy samples, the model required no reflective modifications, indicating first-pass translations meet high-quality standards. For difficult samples, modifications drastically decreased from 295 to 12, a 96% reduction. This demonstrates the model's sensitivity to translation difficulty: for easy and medium samples, the model fully internalized reflective capability, generating high-quality translations during initial translation; for genuinely difficult samples, the model retained reflective improvement capability to ensure translation quality.

4.6 Ablation Study

Table 4: Ablation study on the reflection mechanism evaluated by GRF scores. "*Initial*" and "*Final*" refer to the translations generated before and after explicit reflection. The "*RL*" stage is initialized from the checkpoint of the SFT model.

Model Configuration	Initial	Final
Qwen-7B-Instruct (Baseline)	73.52	–
ReflectMT-SFT (w/o init & refl)	–	75.86
ReflectMT-SFT (w/o refl)	<u>76.85</u>	76.49
ReflectMT-SFT (full)	76.35	77.26
ReflectMT-RL (w/o init & refl)	–	<u>78.62</u>
ReflectMT-RL (full)	**79.73**	**80.51**

To validate the effectiveness of the reflection mechanism, we compare three training configurations, where *init* and *refl* denote initial translation and reflection:

1)ReflectMT (w/o init & refl). The model is trained to map the source sentence directly to the final translation, omitting both the initial translation and the reflection steps.

2)ReflectMT (w/o refl). The model is trained to generate an initial translation followed directly by a refined translation, bypassing the explicit reflection analysis. This variant learns a "*translation-refinement*" process.

3)ReflectMT (full). Our complete configuration, which trains the model on the full "*translation-reflection-refinement*" trajectory.

Table 4 demonstrates that the reflection mechanism significantly enhances translation quality. During the SFT stage, the *w/o refl* variant yields a final GRF score (76.49) lower than its initial translation (76.85), indicating that blind refinement without explicit guidance can degrade performance. Conversely, the *full* model improves from 76.35 to 77.26, proving reflection provides accurate modification directions. Having established the necessity of reflection during SFT, we evaluate the ultimate performance ceiling after RL training. Comparing direct translation (*w/o init & refl*, 78.62) to the full ReflectMT (80.51), the 1.89-point GRF gap confirms the substantial benefit of the complete "*translate-reflect-refine*" paradigm.

5 Related Work

Application of Deep Reasoning Models in Machine Translation. With the emergence of Large Reasoning Models (LRMs) such as OpenAI o1 [16] and DeepSeek-R1 [3], deep reasoning capabilities have pioneered extensive research in long Chain-of-Thought (CoT) reasoning. Recently, researchers have begun exploring their potential in Machine Translation (MT). Zhao et al. [34] and Liu et al. [13] demonstrated the prospects of long CoT reasoning in translation through heuristic examples. Wang et al. [23] constructed the MetaphorTrans dataset and trained the DRT model to handle rhetorical devices in literary translation. DeepTrans-7B [24] leveraged DeepSeek-V3 for reference-free evaluation during Reinforcement Learning (RL) training, exploiting the powerful capability of

large language models as judges. Building on this, Wang et al. [25] proposed the ExTrans model with exemplar-enhanced RL reward modeling, integrating dual capabilities of LLMs as both exemplar generators and judges, achieving breakthroughs in deep reasoning MT and extending to low-resource multilingual translation.

Reflection and Self-Improvement. Reflection and self-improvement have been widely explored to enhance LLM outputs. Shinn et al. [20] proposed Reflexion, which converts task feedback into verbal reflections and stores them in episodic memory to guide subsequent attempts without updating model parameters. Ji et al. [5] explored iterative self-reflection to mitigate hallucinations by checking the factuality of generated knowledge and the consistency of answers. Weng et al. [27] introduced backward verification, treating candidate conclusions as conditions and using verification scores to select reliable answers.

Recent work has further improved the applicability and reliability of reflection. TasTe [26] applies reflection to translation through multi-task supervised instruction tuning for translation, quality prediction, and draft refinement. Its two-stage inference procedure first generates a draft and predicts its quality as a label or score, then refines the translation conditioned on the source text, draft, and quality prediction. IoRT [12] addresses redundancy, drift, and persistent errors in iterative reflection through dynamic meta-instructions. An instructor guided by meta-thoughts and answer-consistency information issues refresh, stop, or select instructions to regulate subsequent iterations. PR-CoT [1] uses prompting to organize structured reflection across logical consistency, information completeness, potential biases, and alternative solutions, then synthesizes these critiques to refine the initial reasoning and answer without additional training.

Unlike approaches that retain explicit reflection or verification at inference time, ReflectMT aims to internalize reflection benefits through two-stage reinforcement learning. In particular, its Stage-2 reward optimization targets first-pass translation rather than relying on assessment-conditioned refinement. This shifts the focus from executing a reflective inference procedure to improving a direct translation policy, without explicit reasoning or iterative revision at deployment.

Knowledge Distillation and Mutual Learning. Knowledge distillation has been explored in MT through confidence-based transfer of bidirectional global-context information [35], joint teacher–student learning [15], and future-context transfer [31]. Related work in code generation exploits complementary AST traversal orders through mutual distillation [28]. These studies leverage additional knowledge during training, while ReflectMT focuses on internalizing reflection through RL for direct translation. SPEC-Distill [11] further suggests a potential direction for reducing our RL rollout costs through speculative prefix reuse, which we leave for future work.

6 Conclusion

This paper introduces a novel "*translate-reflect-refine*" paradigm for machine translation. We design a multi-agent collaborative system to construct reflective translation datasets and propose a two-stage reinforcement learning framework that internalizes reflection capabilities into the model's direct responses. Across our primary English-to-Chinese setting and preliminary evaluations on three additional directions, ReflectMT consistently reduces inference-time output tokens relative to explicit-reflection LRMs while achieving competitive translation quality. Critically, our model generates high-quality translations without explicit reflection steps during inference, avoiding the computational overhead of long COT reasoning while maintaining competitive translation quality. Ablation studies further confirm the effectiveness of the reflection mechanism in our framework.

7 Limitations

While we have demonstrated the effectiveness of our reflection framework across multiple language pairs, several limitations are worth noting. Evaluation on low-resource languages remains limited, and the effectiveness of our reflection mechanism under scarce training data and limited linguistic resources requires further empirical investigation. Moreover, while our model significantly reduces computational overhead during inference, the training pipeline—including multi-agent data construction and two-stage RL—introduces certain upfront computational costs compared to standard fine-tuning.

8 Acknowledgements

The project was supported by National Natural Science Foundation of China (No. 62676342), the Open Competition for Innovative Projects of Xiamen (No. 3502Z20251012), Natural Science Foundation of Fujian Province of China (No. 2024J011001), and the Public Technology Service Platform Project of Xiamen (No.3502Z20231043). We also thank the reviewers for their insightful comments.

References

[1] Mariana Costa, Alberlucia Rafael Soarez, Daniel Kim, and Camila Ferreira. Enhancing self-correction in large language models through multi-perspective reflection, 2026. URL https://arxiv.org/abs/2601.07780.

[2] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[3] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[4] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

[5] Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. Towards mitigating hallucination in large language models via self-reflection, 2023. URL https://arxiv.org/abs/2310.06271.

[6] Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. Metricx-24: The google submission to the wmt 2024 metrics shared task, 2024. URL https://arxiv.org/abs/2410.03983.

[7] Tom Kocmi and Christian Federmann. Large language models are state-of-the-art evaluators of translation quality. In Mary Nurminen, Judith Brenner, Maarit Koponen, Sirkku Latomaa, Mikhail Mikhailov, Frederike Schierl, Tharindu Ranasinghe, Eva Vanmassenhove, Sergi Alvarez Vidal, Nora Aranberri, Mara Nunziatini, Carla Parra Escartín, Mikel Forcada, Maja Popovic, Carolina Scarton, and Helena Moniz, editors, *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland, June 2023. European Association for Machine Translation. URL https://aclanthology.org/2023.eamt-1.19/.

[8] Terry Koo and Mae Li. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, 15, 03 2016. doi: 10.1016/j.jcm.2016.02.012.

[9] Yunlong Liang, Fandong Meng, Jiaan Wang, and Jie Zhou. Slangdit: Benchmarking llms in interpretative slang translation, 2025. URL https://arxiv.org/abs/2505.14181.

[10] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[11] Bingshuai Liu, Zijun Min, Ante Wang, Jiajun Cao, Haibo Zhang, Liang Yao, Shuman Liu, Anxiang Zeng, and Jinsong Su. SPEC-distill: Towards efficient speculative distillation for reinforcement learning with verifiable rewards. In Hung-Yi Lee, Sunipa Dev, Jessy Li, and Desmond Elliott, editors, *Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing*, pages 18946–18959, Budapest, Hungary, October 2026. Association for Computational Linguistics. ISBN 979-8-89176-459-0. doi: 10.18653/2026.emnlp-main.814. URL https://aclanthology.org/2026.emnlp-main.814/.

[12] Liping Liu, Chunhong Zhang, Likang Wu, Chuang Zhao, Zheng Hu, Ming He, and Jianping Fan. Instruct-of-reflection: Enhancing large language models iterative reflection capabilities via dynamic-meta instruction, 2025. URL https://arxiv.org/abs/2503.00902.

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

[14] Qingsong Ma, Johnny Wei, Ondřej Bojar, and Yvette Graham. Results of the wmt19 metrics shared task: Segment-level and strong mt systems pose big challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 62–90, 2019.

[15] Zhongjian Miao, Wen Zhang, Jinsong Su, Xiang Li, Jian Luan, Yidong Chen, Bin Wang, and Min Zhang. Exploring all-in-one knowledge distillation framework for neural machine translation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2929–2940, 2023.

[16] OpenAI. Learning to reason with large language models. https://openai.com/index/learning-to-reason-with-llms/, 2024.

[17] Team Qwen. Qwq: Reflect deeply on the boundaries of the unknown. *Hugging Face*, 2024.

[18] Ricardo Rei, Nuno M. Guerreiro, José Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José G. C. de Souza, and André F. T. Martins. Scaling up cometkiwi: Unbabel-ist 2023 submission for the quality estimation shared task, 2023. URL https://arxiv.org/abs/2309.11925.

[19] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

[20] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

[21] 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. No language left behind: Scaling human-centered machine translation, 2022. URL https://arxiv.org/abs/2207.04672.

[22] Robert J Tibshirani and Bradley Efron. An introduction to the bootstrap. *Monographs on statistics and applied probability*, 57(1):1–436, 1993.

[23] Jiaan Wang, Fandong Meng, Yunlong Liang, and Jie Zhou. Drt-o1: Optimized deep reasoning translation via long chain-of-thought. *arXiv e-prints*, pages arXiv–2412, 2024.

[24] Jiaan Wang, Fandong Meng, and Jie Zhou. Deep reasoning translation via reinforcement learning. *arXiv preprint arXiv:2504.10187*, 2025.

[25] Jiaan Wang, Fandong Meng, and Jie Zhou. Extrans: Multilingual deep reasoning translation via exemplar-enhanced reinforcement learning. *arXiv preprint arXiv:2505.12996*, 2025.

[26] Yutong Wang, Jiali Zeng, Xuebo Liu, Fandong Meng, Jie Zhou, and Min Zhang. Taste: Teaching large language models to translate through self-reflection, 2024. URL https://arxiv.org/abs/2406.08434.

[27] Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large language models are better reasoners with self-verification, 2023. URL https://arxiv.org/abs/2212.09561.

[28] Binbin Xie, Jinsong Su, Yubin Ge, Xiang Li, Jianwei Cui, Junfeng Yao, and Bin Wang. Improving tree-structured decoder training for code generation via mutual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14121–14128, 2021.

[29] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[30] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[31] Biao Zhang, Deyi Xiong, Jinsong Su, and Jiebo Luo. Future-aware knowledge distillation for neural machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(12): 2278–2287, 2019.

[32] Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. AdaptThink: Reasoning models can learn when to think. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 3716–3730, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.184. URL https://aclanthology.org/2025.emnlp-main.184/.

[33] YuJian Zhang, Keyu Chen, Zhifeng Shen, Ruizhi Qiao, and Xing Sun. Adaptive dual reasoner: Large reasoning models can think efficiently by hybrid reasoning. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025. URL https://openreview.net/forum?id=k0YBIBkDKs.

[34] Yu Zhao, Huifeng Yin, Bo Zeng, Hao Wang, Tianqi Shi, Chenyang Lyu, Longyue Wang, Weihua Luo, and Kaifu Zhang. Marco-o1: Towards open reasoning models for open-ended solutions. *arXiv preprint arXiv:2411.14405*, 2024.

[35] Chulun Zhou, Fandong Meng, Jie Zhou, Min Zhang, Hongji Wang, and Jinsong Su. Confidence based bidirectional global context aware training framework for neural machine translation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2878–2889, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.206. URL https://aclanthology.org/2022.acl-long.206/.

A Our Data

Our dataset is constructed by randomly sampling from WMT19 [14], SlangDIT [9], and Metaphor-Trans [23]. We employ DeepSeek-V3 to generate the corresponding translations and reflections. Subsequently, we categorize the difficulty of these samples based on their final GRF scores and filter the data accordingly. The resulting dataset maintains an approximate 1:1 ratio of simple to difficult samples. The detailed dataset composition and the numbers of training, validation, and test examples from each source are reported in Table 11 and Table 5, respectively, while Table 6 presents the distribution of reflection rounds in the training set. By incorporating these difficult samples, we aim to fully elicit the model's reasoning capabilities, thereby encouraging it to learn in-depth reflection and idiomatic translation.

During the Cold Start Supervised Fine-Tuning (SFT) stage, we train the model using the complete samples—comprising the initial translation, reflection analysis, and final translation—to help it internalize the entire reflective translation pipeline. Specifically, rather than retaining only the last refinement transition, we place the initial translation y_0 in `<answer>`, use an LLM to consolidate the feedback and scores from subsequent rounds into a single comprehensive `<reflection>`, and place the final translation y_K in `<final_answer>`. Each SFT instance therefore captures the trajectory from the initial draft to the final refinement, even when the data-construction loop involves multiple reflection rounds. Conversely, in the Reinforcement Learning (RL) stage, we provide only the source English sentences as input, without any reference translations or annotations. For data generation, we locally deploy DeepSeek-V3 (671B) with the temperature set to 0.1 and top-p to 0.95. To ensure data quality, we use rule-based rejection sampling that combines regular-expression format checks with predefined rejection criteria applied to LLM Reflector scores, rather than relying on an LLM to make filtering decisions directly. Furthermore, the hyperparameters in Section 2(3) are set as follows: $\theta = 90$, $K_{max} = 5$, $r_{low} = 40$.

The data format for cold start SFT is as follows:

<answer>2018年非洲面临的三大人道主义挑战</answer> <reflection>1. 翻译质量分析：- 准确性：翻译准确传达了原文的核心信息，"Three Humanitarian Challenges"被准确地译为"三大人道主义挑战"，"for Africa in 2018"也被正确地处理为"2018年非洲面临的"。 - 流畅性：译文符合中文表达习惯，语句通顺自然。 - 语境适配：译文很好地适应了新闻报道或学术文章的语境，保持了正式和客观的语气。 - 过度翻译：未发现过度翻译现象，译文简洁明了。2. 存在的问题： - 无显著问题。译文在各方面都达到了较高的标准。3. 改进建议： - 虽然译文已经很完美，但可以考虑在特定语境下增加一些修饰词以增强感染力，例如："2018年非洲亟待解决的三大紧迫人道主义挑战"。这种调整可以根据具体使用场景灵活选择。"</reflection> <need_revision>否</need_revision> <final_answer>2018年非洲面临的三大人道主义挑战</final_answer>

The initial format of our reflective translation dataset is as follows:

{"src": "The big losers from this economic disaster are workers in the advanced countries that bought into the laissez-faire flexibility of American-style capitalism.",
"ref": "这场经济灾难的大输家是那些认同美式资本主义放任自流政策的发达国家的工人。",
"final_translation": "在这场经济灾难中遭受重大损失的是那些接受了美国式资本主义自由放任灵活性的发达国家的工人。",
"total_rounds": 1,
"final_score": 90,
"best_round": 1,
"history": [{"round": 1,
"translation": "在这场经济灾难中遭受重大损失的是那些信奉美国式资本主义自由放任灵活性的发达国家的工人。",
"reflection": "1. 翻译质量分析： - 准确性：翻译基本准确传达了原文的意思，特别是“economic disaster”（经济灾难）、“big losers”（遭受重大损失）和“American-style capitalism”（美国式资本主义）等关键术语的处理得当。 - 流畅性：译文整体较为流畅，符合中文表达习惯。句子结构清晰，逻辑连贯。 - 语境适配：译文很好地适应了原文的经济学背景，尤其是对“laissez-faire flexibility”（自由放任灵活性）的解释和处理，符合经济学领域的常用表达。 - 过度翻译：没有明显的过度翻译现象，所有信息点都得到了恰当的传达。2. 存在的问题： - “bought into”这一短语的翻译略显生硬，直译为“信奉”虽然可以理解，但可能不够精准。“bought into”在这里更强调对这些理念的接受或认同，而非单纯的“信奉”。3. 改进建议： - 可以将“bought into”翻译为“接受了”或“认同了”，以更准确地传达原文的含义。例如：“在这场经济灾难中遭受重大损失的是那些接受了美国式资本主义自由放任灵活性的发达国家的工人。",
"score": 90}]}}

Table 5: Per-source dataset statistics.

Source	Train	Validation	Test
WMT19	5,482	400	866
SlangDIT	2,856	259	497
MetaphorTrans	3,051	341	637
Total	11,389	1,000	2,000

Table 6: Distribution of `total_rounds` in the training set.

total_rounds	Instances	Percentage
1	3,443	30.23%
2	5,425	47.63%
3	778	6.83%
4	362	3.18%
5	1,381	12.13%

Initial-Translator Robustness. Both roles use DeepSeek-V3 in our default data-construction pipeline. As a model-separation check, we replace only the initial Translator with Qwen-72B while keeping DeepSeek-V3 as the Reflector. Table 7 reports the GRF scores of the downstream cold-start SFT models under both configurations. This comparison provides preliminary evidence of robustness to the initial-translator choice, rather than a claim that using the same model is universally optimal.

Table 7: Effect of the initial-translator choice on downstream cold-start SFT performance. Translator and Reflector denote the models used for initial translation and reflection during data construction, respectively.

Translator	Reflector	SFT GRF
DeepSeek-V3	DeepSeek-V3	76.35
Qwen-72B	DeepSeek-V3	75.83

Table 8: Per-annotator mean scores for the human evaluation in Section 4.4.

Annotator	DeepSeek-R1 (w/ refl)	ReflectMT
1	90.82	91.40
2	89.15	91.28

Table 9: Reward-judge robustness. The first two rows are reproduced from Table 1; the last row replaces DeepSeek-V3 with Qwen-72B as the reward judge during RL training.

	Our				WMT23				WMT24				FLORES-200			
Model	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
DeepSeek-R1 (w/refl)	81.15	2.50	76.67	659.83	89.79	1.93	81.04	1029.68	85.61	2.56	79.28	1069.07	90.50	1.56	84.80	982.33
ReflectMT-Stage2 (Our)	82.55	2.00	78.05	30.73	89.48	1.70	81.18	30.67	86.16	2.10	79.86	47.57	91.42	1.33	84.72	29.10
ReflectMT-Stage2 (Qwen-72B reward)	81.91	2.26	78.92	28.94	88.72	1.98	81.11	29.63	85.63	2.41	79.83	49.16	90.97	1.53	84.51	29.95

B Human Evaluation Details

Table 8 provides the per-annotator mean scores underlying the aggregate results in Section 4.4. Inter-annotator agreement, measured using the two-way random-effects, absolute-agreement intraclass correlation coefficient for average ratings (ICC(2,k)) [8], was 0.58, indicating moderate agreement.

C Robustness Analysis

Effect of Reward Judge Choice. To examine reward-judge dependence, we repeated RL training with Qwen-72B replacing DeepSeek-V3 as the reward judge. Table 9 compares the resulting Stage-2 model with the model trained using the original reward judge and the DeepSeek-R1 baseline with reflection. These results suggest that the efficiency–quality trade-off is not solely tied to the DeepSeek-V3 reward judge, while showing some quality variation across judge choices.

Multi-Seed RL Results. A single RL run may not provide sufficient evidence of a reliable performance advantage. We therefore conducted two additional RL runs with different random seeds during the RL stage. Together with the original run, this yields three independent runs with WMT24 GRF scores of 86.16, 86.05, and 85.97, corresponding to a mean of 86.06 and a sample standard deviation of 0.10. For comparison, Qwen3-8B with reflection and DeepSeek-R1 with reflection achieve WMT24 GRF scores of 85.92 and 85.61, respectively. These results suggest that our method consistently achieves competitive translation quality compared with existing large reasoning models (LRMs), rather than relying on a favorable outcome from a single run.

Statistical Significance Tests. We selected DeepSeek-R1 with reflection as a representative large reasoning model (LRM) baseline for statistical comparison. To assess whether the observed performance differences are statistically significant, we conducted paired bootstrap tests [22] on WMT24 ($N = 998$) and our evaluation dataset ($N = 2000$). As shown in Table 10, the GRF differences are statistically significant on both datasets, with $p = 0.015$ on WMT24 and $p = 0.0053$ on our evaluation dataset. The CometKiwi (CK) differences are also significant on both datasets ($p < 0.001$). These results provide statistical support for the observed performance differences relative to DeepSeek-R1 with reflection.

D Evaluation Prompt

The evaluation prompt of GRF borrows from Kocmi and Federmann [7], and is also employed in Wang et al. [23, 24, 25]:

Table 10: Paired bootstrap tests against DeepSeek-R1 with reflection. Significance levels: * p < 0.05, ** p < 0.01, and *** p < 0.001.

Dataset	Baseline	Metric	N	p_{boot}	Sig.
WMT24	DeepSeek-R1 w/ refl	GRF	998	0.015	*
WMT24	DeepSeek-R1 w/ refl	CK	998	< 0.001	***
Our	DeepSeek-R1 w/ refl	GRF	2000	0.0053	**
Our	DeepSeek-R1 w/ refl	CK	2000	< 0.001	***

Score the following translation from `[src lang]` to `[trg lang]` on a continuous scale from 0 to 100, where score of zero means "no meaning preserved" and score of one hundred means "perfect preservation of meaning, with faithfulness, expressiveness, and elegance".

`[src lang]` source: {src}
`[trg lang]` translation: {hyp}

Score:

The prompt evaluates a translation from a general perspective, and achieves a high correlation with humans [7].

E Data Construction Prompts

The Translator uses a minimal translation instruction without a specialized persona; translation-specific feedback is provided through the Reflector and refinement instructions. The exact prompts for all three steps, including the required output tags, are provided below.

Translator Prompt.

Please translate the following English text into Chinese. Only provide the translation result:
Source text: src_text
Please return your answer in the following format:
<answer>translation</answer>

Reflector Prompt.

Please reflect on and evaluate the following English-to-Chinese translation:
Source text (English): src_text
Current translation (Chinese): translation
Previous reflection: previous_reflection
Please evaluate the translation from the perspectives of accuracy, fluency, and contextual adequacy, and provide suggestions for improvement along with a score.
Please strictly return your answer in the following format: <reflect>
Detailed reflection, including:
1. Translation quality analysis (accuracy, fluency, contextual adequacy, whether there is over-translation)
2. Existing problems (if any)
3. Suggestions for improvement (if any)
</reflect>
<score>score (an integer between 0 and 100)</score>

Refinement Prompt.

Table 11: Dataset statistics for translation tasks.

Split	EN→ZH	ZH→EN
Train	11,389	8,000
Test	2,000	2,000
Val	1,000	1,000

Please improve the following English-to-Chinese translation based on the reflection:
Source text (English): src_text
Current translation (Chinese): previous_translation
Reflection: reflection
Please provide the improved translation in the following format:
<answer>improved translation</answer>

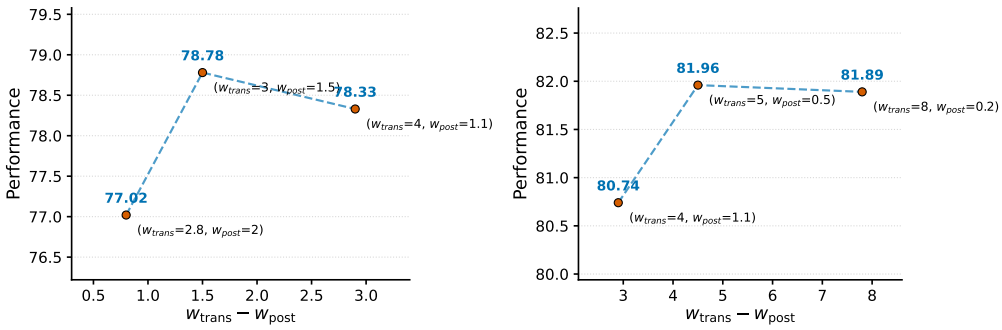
F Reflection Quality Evaluation Prompt

We use DeepSeek-V3 to evaluate reflection quality based on accurate issue identification, in-depth analysis, justified alternatives, and coherent reasoning. The exact evaluation prompt is provided below, and the resulting score on a 0–100 scale is normalized to $[0, 1]$ to serve as the reflection reward.

You are an expert in evaluating translation reflection processes. A translation task requires translating the following English text into Chinese:
<source>source</source>
Someone provided an initial translation:
<initial_translation>initial_translation</initial_translation> Then they reflected on their translation:
<reflection>reflection</reflection>
Please evaluate the quality of the reflection process on a scale of 0 to 100, where:
- **0 points**: Empty reflection or less than 20 characters; no meaningful content.
- **1-10 points**: Extremely superficial reflection; only generic statements like "translation is correct" with no actual analysis.
- **11-30 points**: Basic reflection; mentions some issues but lacks depth; no discussion of nuances, context, or alternative strategies.
- **31-50 points**: Moderate reflection; identifies some translation issues and provides basic analysis; mentions alternative approaches but without detailed justification.
- **51-70 points**: Good reflection; demonstrates deep analysis of translation issues, context, and cultural nuances; discusses alternative strategies with reasonable justification.
- **71-100 points**: Excellent reflection; provides comprehensive analysis of translation problems, explores multiple alternatives with detailed justification; shows deep understanding of both languages and cultural contexts.
Evaluation Criteria:
1. **Problem Identification** (30%): Does it accurately identify issues in the initial translation?
2. **Analysis Depth** (30%): Does it deeply analyze why the initial translation has problems?
3. **Alternative Solutions** (25%): Does it propose better translation alternatives?
4. **Logical Coherence** (15%): Is the reflection process clear and well-organized?
Only output a score.
score: Your score

G Implementation Details

Cold Start SFT. We employed the LLaMA-Factory framework for Supervised Fine-Tuning (SFT). The model was trained for 3 epochs with a learning rate of 1×10^{-4} and a batch size of 1 with 8



(a) Stage 1 (b) Stage 2

Figure 4: Effect of the reward-weight gap $w_{\text{trans}} - w_{\text{post}}$ on the two training stages. Scores are reported on the validation set.

Table 12: Experimental results in Chinese-to-English translation. **Bold** and underlined values denote the best and second-best scores, respectively.

Model	Our				WMT23				WMT24				FLORES-200			
	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
Qwen-7B-Instruct	73.16	2.20	78.36	21.31	82.00	2.54	78.11	34.40	83.83	2.49	80.05	44.32	84.38	1.59	80.61	28.64
Qwen2.5-14B	79.89	2.30	79.97	26.00	87.27	2.46	80.46	46.32	86.35	2.58	81.66	55.92	88.98	1.83	85.24	32.45
Qwen2.5-14B(w/ refl)	86.91	1.67	80.78	183.72	90.65	1.72	81.15	210.63	90.09	2.15	81.96	225.70	92.91	0.99	85.77	641.89
Llama3.1-8B	75.57	2.25	79.98	25.87	79.86	2.60	79.56	37.53	81.77	2.68	80.44	57.79	84.12	1.81	83.53	27.23
Qwen3-8B	88.24	1.59	80.93	566.54	90.23	1.64	80.67	561.76	89.88	1.98	81.95	593.48	93.80	0.98	84.29	580.23
Qwen3-8B(w/ refl)	89.97	1.57	81.29	669.21	92.03	1.63	81.38	698.51	90.77	1.97	**82.41**	758.57	95.14	0.98	85.77	681.00
Marco-o1	80.24	1.81	80.35	490.37	87.70	2.01	80.34	498.71	86.15	2.28	80.56	495.59	90.20	1.30	85.31	477.49
Marco-o1(w/ refl)	86.08	1.68	80.44	602.75	88.54	1.77	80.54	574.07	87.65	2.09	81.09	650.91	93.03	1.05	85.33	584.38
DeepSeek-R1	87.29	1.62	80.67	411.58	91.07	1.54	80.71	431.80	90.36	1.97	81.91	542.17	93.32	0.94	85.86	526.94
DeepSeek-R1(w/ refl)	90.02	1.59	81.35	1567.80	**92.26**	1.53	**81.59**	1631.93	**91.39**	**1.94**	82.09	1542.51	94.64	0.94	**85.97**	1526.89
QwQ-32B	87.01	1.53	80.41	602.75	87.25	1.60	81.32	633.19	88.96	2.02	81.04	645.55	91.31	1.05	80.40	627.47
QwQ-32B(w/ refl)	88.67	1.51	80.46	1000.16	89.09	1.58	81.38	1056.95	91.34	2.02	81.78	1065.69	92.84	1.04	84.09	949.07
Qwen-7B-LoRA-v3(Cold Start)	86.78	1.63	80.86	**20.43**	88.59	1.75	80.75	**32.45**	87.23	2.88	80.31	**40.95**	93.64	1.06	84.93	27.22
Qwen-7B-RL-v3(Our)	88.88	1.58	81.22	23.71	89.94	1.59	81.27	34.91	89.71	2.51	81.98	41.46	94.36	1.01	85.22	**26.87**
Qwen-7B-RL-v3-Round2(Our)	**90.35**	**1.50**	**81.38**	22.78	91.13	**1.52**	81.49	34.50	90.07	2.11	82.32	44.33	**95.56**	**0.93**	85.82	29.16

gradient accumulation steps. The entire SFT process required approximately 3 GPU hours (equivalent NVIDIA A100 40G GPU hours).

RL Training. We implemented the GRPO algorithm using the `verl`[6] library. We locally deployed DeepSeek-V3 to serve as the reward model for optimizing the policy model. For the training configuration, we set the global batch size to 64, the learning rate to 2×10^{-7}, the number of rollouts to 8, the rollout temperature to 0.7, and the KL coefficient to 5×10^{-3}. The training was conducted for 2 epochs, consuming a total of approximately 770 GPU hours (equivalent NVIDIA A100 40G GPU hours).

Hyperparameters. In Stage 1, the reward weights were set as follows: $w_{\text{trans}} = 3$ and $w_{\text{post}} = 1.5$. In Stage 2, we adjusted these weights to $w_{\text{trans}} = 5$ and $w_{\text{post}} = 0.5$. These values are selected based on the ablation in Figure 4(a) and Figure 4(b), where each configuration achieves the best performance in its respective stage.

Evaluation. To evaluate the baseline models, we obtained open-source models from Hugging Face and accessed closed-source models via their official APIs. We utilized the official implementations and model checkpoints to compute the COMETKIWI-XL and MetricX metrics. Similarly, the GRF scores were obtained via API calls.

H Reflection Consolidation Prompt

The following prompt consolidates multiple rounds of translation feedback into a single, coherent reflection that directly guides the revision from the initial translation to the final translation.

[6] `https://github.com/volcengine/verl`

You are a translation revision editor. Your task is to consolidate multiple rounds of translation reflections into a unified set of revision feedback that can be used directly to revise the "Initial Translation" into the "Final Translation".

Input

<initial_translation> initial_translation </initial_translation>
<intermediate_reflections> intermediate_reflections </intermediate_reflections>
<final_translation> final_translation </final_translation>

Task Objective

Generate an "Aggregated Reflection" as the intermediate step in the following training sequence:

Initial Translation $\rightarrow$ Aggregated Reflection $\rightarrow$ Final Translation

The aggregated reflection must directly address the initial translation, identify specific issues that need revision, and provide revision directions consistent with the final translation. It should be standalone, coherent, and actionable revision feedback, rather than a summary of the multi-round reflections or a retrospective account of the revision process.

Consolidation Rules

1. Use the initial translation as the sole reference for locating issues - Every issue must correspond to something present in the initial translation. - Remove issues that apply only to intermediate versions and do not exist in the initial translation. - If an issue underwent multiple rounds of revision, consolidate them into a recommendation that leads directly from the initial translation to the final translation, without retaining intermediate attempts.
2. Use the final translation as the target version - Retain reflections that explain or guide the actual changes from the initial translation to the final translation. - Remove suggestions that were not adopted in the final translation, were overturned by subsequent reflections, or conflict with the final translation. - When suggestions from different rounds disagree, retain only the parts that are consistent with the final translation and supported by the input. - The final translation is the target version for this task, but this does not mean that every change necessarily corrects an objective error. Changes involving only wording or stylistic preferences should be described as refinements in expression, not overstated as corrections of mistranslations.
3. Merge duplicate content - If the same issue appears in multiple rounds of reflection, include it only once. - Combine complementary information into complete, concise recommendations. - Do not retain narratives of repeated experimentation, withdrawn suggestions, or the evolution of different versions.
4. Make recommendations specific and actionable - Each recommendation should, wherever possible, identify the specific location or wording in the initial translation, the aspect that needs adjustment, and a clear direction for revision. - Quote brief excerpts or provide local replacement wording when necessary to eliminate ambiguity. - Avoid vague comments such as "improve accuracy" or "make the wording more natural" without identifying a specific target. - If a change affects multiple locations, specify the scope of the consistent revision, such as terminology, forms of address, tense, or register.
5. Cover key changes without fabricating justifications - Prioritize key changes involving meaning, completeness of information, terminology, logical relationships, grammar, tone, and style. - If the multi-round reflections omit important changes between the initial and final translations, you may add revision recommendations directly supported by the two versions. - The input does not include the source text. Do not fabricate source content or present unsupported judgments as "The source text explicitly states..." - If you cannot determine whether a change corrects a mistranslation, describe only the observable adjustment in expression without inventing a semantic justification. - Do not force an explanation for differences that cannot be explained by the input.

Output Format

Output only the body of the aggregated reflection. Do not include task instructions, the analysis process, version comparison tables, or any other additional content.

Table 13: Experimental results in English-to-German translation. **Bold** and underlined values denote the best and second-best scores, respectively.

Model	WMT23				WMT24				FLORES-200			
	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
Qwen-7B-Instruct	59.04	4.07	74.01	142.50	57.25	3.55	74.89	63.34	54.10	2.36	79.21	42.36
Qwen2.5-14B	72.91	2.82	78.12	139.04	67.41	2.45	79.50	75.30	77.52	1.07	83.80	48.24
Qwen2.5-14B(w/ refl)	78.84	2.77	79.08	397.14	69.54	2.19	80.09	284.27	78.32	1.00	84.53	224.11
Llama3.1-8B	70.11	3.11	74.35	**112.08**	57.66	2.42	74.61	69.48	72.28	1.01	84.63	40.18
Qwen3-8B	69.82	2.61	79.93	716.42	73.01	2.27	80.87	491.98	76.74	0.86	84.93	592.83
Qwen3-8B(w/ refl)	70.56	2.64	80.25	860.14	75.32	2.18	80.91	655.36	78.66	0.80	85.67	689.12
Marco-o1	65.51	3.28	76.96	616.28	59.98	2.81	77.52	466.15	61.69	1.39	82.54	345.46
Marco-o1(w/ refl)	66.40	3.23	77.19	849.95	62.64	2.56	78.60	618.26	64.82	1.26	83.21	639.70
DeepSeek-R1	79.42	2.15	81.66	820.33	75.80	1.73	80.83	690.31	73.33	0.61	86.71	840.02
DeepSeek-R1(w/ refl)	80.13	2.05	81.90	1360.79	**78.66**	1.69	**81.77**	1060.92	**79.32**	**0.60**	**86.75**	1139.90
Qwen-7B-LoRA-v3(Cold Start)	73.67	3.71	74.98	124.09	62.04	3.25	75.19	65.67	70.23	1.48	81.36	**40.10**
Qwen-7B-RL-v3(Our)	75.39	2.63	78.37	130.92	64.15	2.49	77.53	64.28	72.03	1.14	83.97	43.82
Qwen-7B-RL-v3-Round2(Our)	**80.83**	**2.00**	**82.51**	119.14	77.10	**1.38**	80.82	**59.49**	78.46	0.68	86.22	40.56

I Our Adapt thinking Baseline

Our pre-thinking baseline uses training data aligned with our method, employing DeepSeek-V3 to generate pre-thinking data for SFT cold start. The data format is as follows:

<think>首先，"kind of"是一个常见的俚语表达，通常用来表示某种程度的不确定性或轻微的程度。然后，"Riverdale"是一个专有名词，但在这个上下文中，它可能指的是一个地方名，也可能是电视剧《Riverdale》中的Riverdale镇。如果是指地点，那么“his last meal in Riverdale”可以理解为他在Riverdale的最后一次用餐；如果是指电视剧，则可能是在描述剧中人物即将离开或结束在Riverdale的生活。考虑到俚语的使用，这句句子更有可能出自一种非正式的语境，比如电视剧的讨论或者个人的情感表达。因此，我将假设此句出自电视剧的讨论背景。</think><answer>有点希望这是他在Riverdale的最后一顿饭</answer>

In datasets where no reflection is required, the ‘<think></think>‘ tag remains empty. The ratio of reflective to non-reflective data is set at 2:3. Using LLaMA-Factory for fine-tuning and verl for GRPO training, we cultivated a model capable of adaptively deciding whether to engage in reflection before generating a response. This adaptive reflection method allows the model to avoid excessive contemplation on simple tasks, thereby preventing performance degradation. Furthermore, it effectively reduces token consumption, enhancing overall efficiency.

J Multilingual

Across Chinese-to-English, English-to-Russian, and English-to-German translation, ReflectMT substantially reduces generated tokens compared with explicit-reflection baselines while achieving competitive translation quality. Detailed results for the three translation directions are reported in Tables 12, 14, and 13, respectively.

K Cultural Adaptation in Reflection

The reflector does not follow a fixed rule of preserving, omitting, or replacing culturally specific expressions. Our qualitative inspection of culturally marked examples sampled from SlangDIT and MetaphorTrans suggests that it generally identifies the expression’s pragmatic function and favors a natural target-language paraphrase or functional equivalent. Source-culture references may be retained when understandable, while the intended strategy avoids omitting source meaning or introducing unsupported target-culture content.

Case Study: “Wigs on the green.” This British idiom refers to an intense dispute or chaotic confrontation. In the inspected example, the reflector judges the initial Chinese rendering “大闹” to be overly colloquial and insufficiently expressive of the intended implication, and subsequently revises it to “陷入一片混乱之中.” The translation quality score increases from 65 to 85. Rather

Table 14: Experimental results in English-to-Russian translation. **Bold** and underlined values denote the best and second-best scores, respectively.

Model	WMT23				WMT24				FLORES-200			
	GRF	MX24	CK	Token	GRF	MX24	CK	Token	GRF	MX24	CK	Token
Qwen-7B-Instruct	63.98	3.90	77.69	52.34	56.07	5.50	73.24	94.69	68.49	3.81	79.13	65.93
Qwen2.5-14B	72.61	3.61	81.31	72.93	61.72	4.07	78.01	91.92	74.27	2.05	83.96	91.36
Qwen2.5-14B(w/ refl)	76.00	2.80	74.43	268.44	65.88	3.84	79.18	276.94	82.92	1.95	85.20	274.99
Llama3.1-8B	70.74	3.87	78.87	64.42	50.98	4.83	74.31	66.77	74.14	2.27	85.09	45.80
Qwen3-8B	74.72	3.10	81.67	487.80	72.59	4.14	79.60	683.68	74.20	2.91	85.38	524.97
Qwen3-8B(w/ refl)	75.09	2.85	82.27	685.75	72.95	4.11	79.66	761.59	75.09	2.80	86.06	634.82
Marco-o1	68.56	3.38	80.29	450.95	61.64	4.68	76.58	591.11	68.29	2.57	83.74	550.43
Marco-o1(w/ refl)	69.20	3.48	79.40	549.94	68.34	4.56	77.48	768.73	73.54	2.46	84.65	601.34
DeepSeek-R1	81.94	2.36	83.38	748.59	81.28	3.16	80.42	985.93	79.82	1.33	87.69	974.30
DeepSeek-R1(w/ refl)	83.07	**2.12**	**84.06**	967.00	**81.50**	**3.15**	**81.00**	1138.65	83.75	1.28	**87.94**	1226.05
Qwen-7B-LoRA-v3(Cold Start)	73.20	3.81	81.58	**37.89**	71.56	4.29	78.08	**64.74**	77.10	2.09	85.74	**44.19**
Qwen-7B-RL-v3(Our)	77.37	2.71	83.14	58.83	75.65	3.55	79.76	74.29	86.76	1.30	85.81	49.24
Qwen-7B-RL-v3-Round2(Our)	**83.75**	**2.12**	83.98	61.78	78.38	3.37	80.23	87.83	**87.01**	**1.26**	87.77	68.76

Table 15: Performance metrics for GPT-4o and ReflectMT-Stage2.

Model	**GRF**	**MX24**	**CK**	**Token**
GPT-4o	77.62	2.63	77.97	28.57
ReflectMT-Stage2 (Our)	82.55	2.00	78.05	30.73

than reproducing the opaque surface form, the revision conveys the expression through a natural target-language paraphrase.

L GPT-4o Score

Due to the costs associated with using the GPT-4o API, we limited our testing of GPT-4o to our custom dataset. As shown in Table 15, ReflectMT achieves higher GRF scores compared to GPT-4o, while maintaining similar token consumption.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in the abstract and introduction are empirically verified in Section 4, in particular Table 1

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We mention our limitations in Section 7.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: This paper does not include any theoretical results or formal proofs; our contribution is empirical, focusing on a training framework and experimental validation.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We describe the full methodology in Section 3, the data construction pipeline in Section 2, and provide detailed hyperparameters, training configurations, backbones, and evaluation prompts in Appendices A–D (e.g., learning rate, batch size, GRPO settings, reward weights, temperature).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code, dataset, training scripts and prompts are publicly released upon publication to ensure full reproducibility.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 describes the datasets, metrics, and backbones. Appendix G provides full implementation details, including SFT hyperparameters, GRPO settings , reward weights for both stages, and evaluation configurations. Dataset splits are given in Table 11.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We conduct paired bootstrap significance tests, report multi-seed RL replicates in Appendix C to assess result stability. We do not plot per-run error bars in main text figures due to high computational cost of full end-to-end RL training. See Appendix C for bootstrap p-values and multi-seed results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix G reports the compute budget in terms of NVIDIA A100 40GB-equivalent GPU hours. Due to institutional policy, we do not disclose the specific GPU model used in our internal cluster. DeepSeek-V3 (671B) is locally deployed and serves as both the data-construction agent and the LLM-as-a-judge reward model.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that our research conforms to it. Our work uses publicly available datasets and does not involve human subjects or sensitive data.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This work focuses on improving the efficiency and quality of machine translation, a well-established and widely-used technology. It does not introduce new capabilities with direct foreseeable negative societal impacts beyond those generally associated with MT systems (e.g., occasional mistranslation).

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: Our released model and dataset target standard English-to-Chinese machine translation and pose no higher risk of misuse than existing publicly available MT models and corpora. Therefore no additional safeguards are required.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All used assets are properly cited, including base models, datasets, and frameworks. We use them in accordance with their licenses and terms of use.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We introduce a new reflective English-to-Chinese translation dataset. Its construction pipeline, statistics, filtering rules, hyperparameters, and representative examples are fully documented in Section 2 and Appendix A. The dataset will be released with documentation upon acceptance.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: This work does not use crowdsourcing or recruit external human-subject participants. Blinded human evaluation reported in Appendix B is completed by internal researchers of our research group who are not listed as paper authors; no paid external annotators are involved.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve research with human subjects; therefore no IRB approval is required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are a core component of our methodology. We use DeepSeek-V3 for reflective data construction (as Translator and Reflector agents) and as the LLM-as-a-Judge reward model during RL training. Qwen2.5-7B-Instruct serves as our base policy model. All usages are described in detail in Sections 2 and 3 and Appendix D.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.