Multi-agentMT: Deploying AI Agent in the WMT25 Shared Task

Ahrii Kim
AI-Bio Convergence Research Institute
South Korea
ahriikim@gmail.com
trotacodigos/MultiAgentMT.git

Abstract

We present Multi-agentMT, our system for the WMT25 General Shared Task. The model adopts Prompt Chaining, a multi-agent workflow combined with RUBRIC-MQM, an automatic MQM-based error annotation metric. Our primary submission follows a **Translate–Postedit–Proofread** pipeline, in which error positions are explicitly marked and iteratively refined. Results suggest that a semi-autonomous agent scheme for machine translation is feasible with a smaller, earlier-generation model in low-resource settings, achieving comparable quality at roughly half the cost of larger systems.

1 Introduction

An AI Agent is a computational system that operates autonomously, guided by environmental observations, and often incorporates adaptive learning abilities (Russell and Norvig, 2010). Recent advances in Large Language Models (LLMs) have greatly enhanced AI Agents by enabling stronger reasoning, contextual understanding, and flexible task execution, particularly in Machine Translation (MT) (Briva-Iglesias, 2025). Building on this progress, Briva-Iglesias (2025) proposed a multi-agent MT system with four agents—Translator, Fluency Reviewer, Adequacy Reviewer, and Editor—which, while still preliminary, demonstrates promising potential. Inspired by this approach, we participate in this year's WMT (Conference on Machine Translation) General Task with an AI multi-agent workflow. **Our objective is to develop a smaller model that surpasses larger counterparts, thereby showcasing the potential of AI Agents in MT while substantially reducing computational cost.**

This year's competition focuses on translating texts across a broad spectrum of languages, domains, genres, and formats. We addressed the **multilingual subtask** covering 30 languages, with

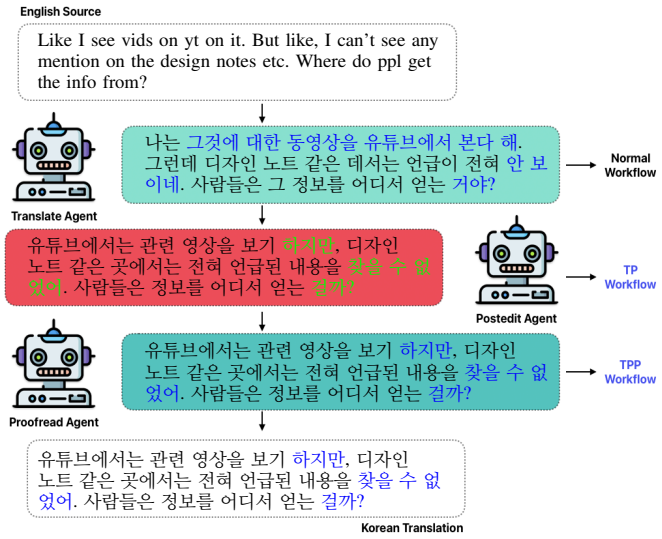

Figure 1: Prompt chaining architecture of Multi-agentMTwith Translate–Postedit–Proofread. Our submission includes two workflows: Translate–Postedit (TP) and Translate–Postedit–Proofread (TPP), with TPP serving as the primary system. In each workflow, agents sequentially process the output of the previous stage, iteratively refining translation quality.

Czech, English, and Japanese as source languages. By adhering to prompt engineering without prioritizing specific languages, our system can be categorized as both a **contrastive** and **unconstrained** model.

One major challenge stemmed from the dataset structure. Following last year's convention, the dataset included document boundaries, with segments composed of multiple sentences or paragraphs separated by one or two newline characters. This format yielded 29,957 segments or 102,060 paragraphs. Our initial submission translated at the segment level but often failed to respect paragraph boundaries—merging or omitting content, particularly within the TPP Workflow (see § 5.2). To address this, we later split segments into individual paragraphs and translated them independently

Proceedings of the Tenth Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 769–777
November 8–9, 2025.

during inference. Apart from this adjustment, most translations were performed at the segment level.

In the official results, our system did not undergo human evaluation because it belonged to the unconstrained category (Kocmi et al., 2025a). Nevertheless, preliminary rankings based on automatic metrics (Kocmi et al., 2025b) suggest that the architecture is particularly effective for low-resource languages. The most notable outcome is observed for English–Serbian, although the underlying factors remain unclear. Considering that our baseline model is not the most up-to-date, this result could be better with other more recent light models in the Multi-agentMT architecture.

The remainder of this paper is organized as follows. Section 2 details the multi-agent architecture. Section 4 presents experimental settings based on the WMT24++ dataset (Deutsch et al., 2025), and Section 5 reports results and analysis. The Appendix provides additional details of the prompt designs.

2 System Overview

2.1 Design

AI Agents enable dynamic workflows through configurable architectures.[16] We adopt the concept of Prompt Chaining, in which each step's output serves as the input for the next, thereby fostering systematic reasoning and iterative refinement (Briva-Iglesias, 2025). While iterative refinement could theoretically improve translation quality, cost considerations led us to adopt a unidirectional configuration. Accordingly, we examine two multi-agent workflows: Translate–Postedit (TP Workflow) and Translate–Postedit–Proofread (TPP Workflow), as illustrated in Figure 1. Both configurations were submitted to the competition.

2.2 Translate Agent

The Translate Agent generates translations of the source text using the official prompt provided by the organizers. Although cost-effective alternatives such as Google Translate or DeepL could be employed, we did not use them as *our preliminary experiments suggested that higher-quality initial translations yielded superior downstream results*.

2.3 Post-edit Agent

The Post-edit Agent revises translations with reference to the source text. It builds on the RUBRIC-MQM framework (Kim, 2025), an LLM-as-judge

Algorithm 1: post_edit_translation(response, tgt_text)

```
Input: response, tgt_text
Output: corrected
raw ← response["content"] or ""
corrected ← tgt_text
MIN_SAFE_SPAN_LEN ← 2
try:
    safe_response ← sanitize_response(raw)
    parsed ← JSON parse of safe_response
    if parsed is a dictionary then
        forall span in parsed do
            info ← parsed[span]
            if info is not a dictionary then
                continue
            suggestion ← clean_suggestion(
                info["suggestion"].strip() )
            if span.lower() == "no-error"
                or suggestion is empty
            or suggestion == span then
                continue
            if length(span) < MIN_SAFE_SPAN_LEN then
                continue
            space ← " "
            pattern_space ← space + escape(span) + space
            (corrected, count) ← regex_subn(pattern_space,
                space + suggestion + space, corrected)
            if count == 0 then
                pattern_general ← escape(span)
                (corrected, _) ← regex_subn(pattern_general,
                    suggestion, corrected)
except:
    corrected ← tgt_text
corrected ← preserve_paragraph(tgt_text, corrected)
return corrected
```

system that classifies MQM-style error categories, severities, and spans, comparable to GEMBA-MQM (Kocmi and Federmann, 2023). RUBRIC-MQM has shown robustness in identifying error categories—especially MAJOR and MISTRANSLATION—and in distinguishing between flawless and flawed sentences.

We revise four aspects of the original framework:

–Error correction Instead of only identifying errors, the model is instructed to propose improved translations for each error span.

–Severity scale The 100-level scale is reduced to 4, as severity is not our primary focus, though Kim (2025) emphasize its importance.

–Multilingual in-context-learning (ICL) examples One English–German example is replaced with a Japanese–Korean one to generalize the framework to X–Y translation directions.

–Mandatory corrections We remove the NO-ERROR option to ensure that at least one correction is proposed. Our preliminary study found that RUBRIC-MQM frequently selected NO-ERROR, leading to no edits throughout the agentic pipeline. When we enforced changes, the model tended to paraphrase rather than leave the sentence unchanged. This behavior aligns with the view that perfect quality is unattainable and any translation can be further improved. To accommodate this, we introduce a new label, STYLE, ensuring the model consistently proposes edits.

As a post-processing step, suggested translations are integrated using Algorithm 1, which applies two substitution strategies:

–Space-sensitive substitution Replaces spans only when surrounded by spaces to avoid partial-word errors.

–Fallback substitution If no replacement occurs, substitutes the span wherever it appears.

This procedure ensures accurate yet comprehensive corrections. The revised sentence constitutes the final output of the TP Workflow.

2.4 Proofread Agent

The Proofread Agent further refines translations using Chain-of-Thought (CoT) prompting (Wei et al., 2022). The model first identifies potential errors, then proposes five fluent alternatives aligned with the source text, and finally selects the most suitable version. This stage is designed to address awkward expressions introduced during earlier revisions. Nevertheless, the agent occasionally produces hallucinations. To mitigate this, we add an additional instruction emphasizing faithfulness to the given translation, which alleviates the issue in many cases. Despite this safeguard, hallucinations may still occur and will require separate verification. The resulting translation constitutes the final output of the TPP Workflow.

3 Performance

In this section, we present the performance of Multi-agentMT under the submitted configuration.

3.1 Model Architecture

All agents are based on GPT-4o-mini (`4o-mini-2024-07-18`), a proprietary OpenAI model (OpenAI et al., 2023), configured with temperature = 1 and max_tokens = 1024. Although this temperature is not optimal for reproducibility, iterative pilot studies suggested that it encouraged broader exploration of errors and corrections, thereby improving performance. The system was executed between June 19 and July 3, 2025. Future work should aim to establish a more stable and reproducible environment.

3.2 Official Result

Since our submission did not undergo human evaluation, the official rankings are based on automatic metrics. We approximate relative performance against other unconstrained models using AutoRank obtained from Kocmi et al. (2025b), following Equation 1.

$$\text{Relative Performance} = \left(1 - \frac{N_{\text{loss}}}{N_{\text{total}}}\right) \times 100 \quad (1)$$

As human scores are not available for these systems, **the results should be interpreted as indicative rather than conclusive, and ultimately require validation through human assessment**.

Figure 2 illustrates Multi-agentMT's relative ranking compared to models that it surpassed at least once across the 31 language pairs. Notably, our system consistently outperformed Online-G, and frequently exceeded TowerPlus-72B and EuroLLM-22B-pre. Figure 3 further shows that Multi-agentMT achieved its best relative position

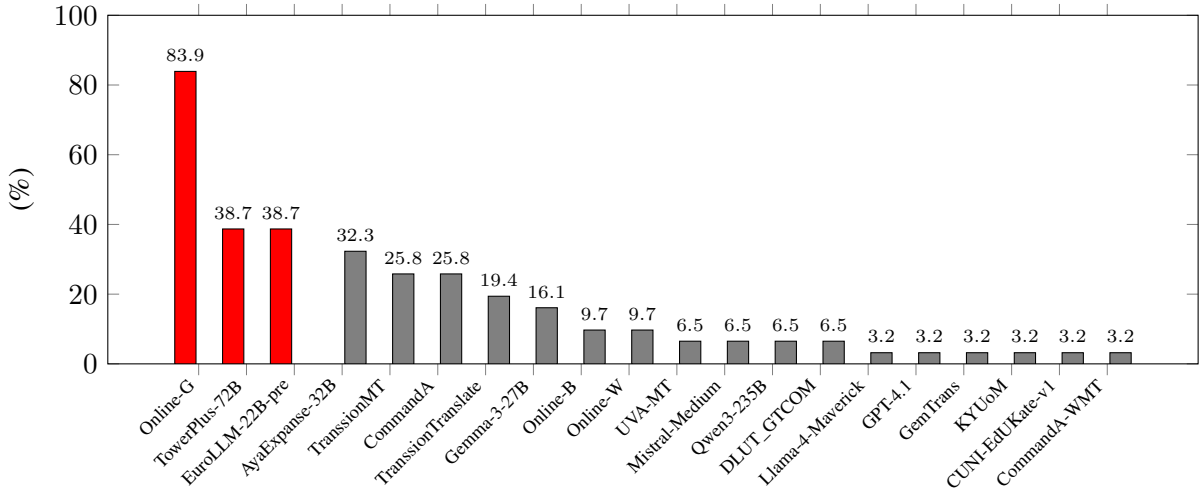

Figure 2: Relative performance of Multi-agentMT in 31 language pairs to unconstrained models. Top-3 models are highlighted in red, others in gray.

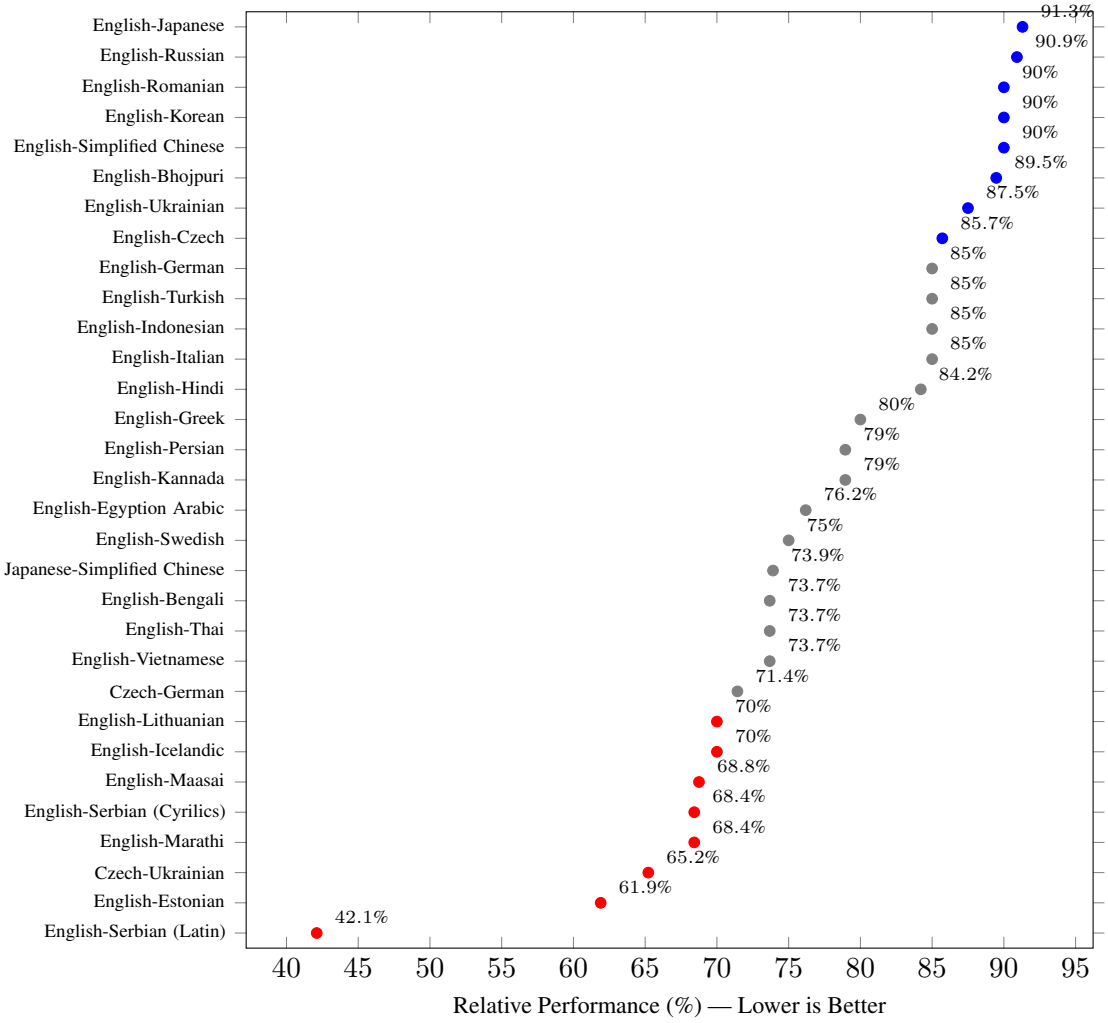

Figure 3: Relative performance of Multi-agentMT across language pairs (bottom 25%, middle 50%, top 25%).

Method	Translate Tokens	Post-edit Tokens	Proofread Tokens	Cost ($)
Translate	10,728,181	–	–	6.05
Post-edit	–	58,548,177	–	33.02
Proofread	–	–	15,874,680	8.95
TP (Translate + Post-edit)	10,728,181	58,548,177	–	39.07
TPP (Translate + Post-edit + Proofread)	10,728,181	58,548,177	15,874,680	48.03
GPT-4o	10,728,181⋆	–	–	100.84
GPT-4.1	10,728,181⋆	–	–	81.53

Table 1: Token usage and cost comparison between our workflows (TP, TPP) and larger GPT models. Assuming that the larger models consume a similar number of tokens (marked with ⋆), our workflows use more tokens but incur lower costs, achieving comparable translation quality.

	4o-mini	4o	4.1
Input	$0.15	$2.50	$3.00
Output	$0.60	$10.00	$8.00

Table 2: API pricing per 1M input/output tokens for various GPT models (OpenAI).

in English–Serbian (Latin), ranking within the top 42.1%. Beyond this case, the system achieved top-25% performance primarily in English-to-low-resource-language directions, suggesting that its robustness is particularly evident under low-resource conditions.

3.3 Cost-efficiency

Table 1 summarizes token usage, and Table 2 shows the pricing structure for each model. Our system exhibits an average input–output ratio of $0.08 : 0.92$, which forms the basis for cost estimation. Notably, the Postedit Agent accounts for 68.75% of total tokens, corresponding to $33.02 of the overall $48.03 expenditure. Assuming comparable token usage to larger models, the results suggest that comparable quality can be achieved in certain languages at roughly half the cost of GPT-4o and 60% of GPT-4.1.

4 Experiment

This section evaluates the relative effectiveness of our model—a compact, earlier-generation variant—on two directions discussed above: English–Serbian and English–Icelandic. We use the WMT24++ dataset (Deutsch et al., 2025), which provides an English source and 55 target-language translations, together with up to two references (a human translation and a post-edited version). After filtering low-quality segments using COMET, we retain 960 source segments per language pair.

Translations are generated with the TP and TPP workflows. We report BLEU (Papineni et al., 2002), ChrF (Popović, 2015), and TER (Snover et al., 2006) using SacreBLEU (Post, 2018); we also report COMET (Rei et al., 2020) with both references and the reference-free COMETKiwi (Rei et al., 2022). To quantify the magnitude of edits introduced by each workflow, we additionally compute TER between the TP and TPP outputs. For cost-efficiency, we record token counts for each setting.

For efficiency, we replace the Translate Agent with off-the-shelf translations from Gemini-1.5-Flash (rather than generating outputs with GPT-4o-mini as in our submission), and set the decoding temperature to 0 for reproducibility.

5 Result

5.1 Performance

As shown in Table 3, metric scores generally decrease after post-editing and subsequently increase after proofreading. Overall, n-gram-based metrics show little to no improvement across stages, while COMET scores improve in both language pairs. This trend suggests that Multi-agentMT introduces beneficial edits by altering vocabulary while largely preserving sentence structure.

To further assess the direct influence of the Postedit Agent, we evaluate translations from the Translate–Proofread pipeline. Table 3 indicates that when the Postedit Agent is omitted, surface-level scores increase but semantic-level scores decline. This implies that the Postedit Agent induces more substantial edits, leading to structural and semantic divergence, which does not necessarily yield positive outcomes.

We next compute TER between workflow stages to quantify the magnitude of edits. As shown in

Language	Metric	Translate	Postedit	Proofread	w/ PE	w/o PE
Icelandic	BLEU	18.33	17.91 (-0.42)	18.00 (+0.09)	18.00 (-0.33)	19.19 (+0.86)
	ChrF	43.42	42.96 (-0.46)	43.55 (+0.59)	43.55 (-0.13)	43.80 (+0.38)
	TER	67.49	69.61 (+2.12)	70.28 (+0.67)	70.28 (+2.79)	72.86 (+5.37)
	COMET	78.75	76.90 (-1.85)	79.22 (+2.32)	79.22 (+0.47)	75.12 (-3.63)
	COMET Kiwi	75.74	73.89 (-1.85)	76.41 (+2.52)	76.41 (+0.67)	73.33 (-2.41)
Serbian	BLEU	23.12	21.92 (-1.20)	20.39 (-1.53)	20.39 (-2.73)	26.01 (+2.95)
	ChrF	49.96	48.26 (-1.70)	46.02 (-2.24)	46.02 (-3.94)	51.13 (+1.17)
	TER	63.79	67.10 (+3.31)	69.73 (+2.63)	69.73 (+5.94)	75.22 (+11.34)
	COMET	82.49	79.31 (-3.18)	81.42 (+2.11)	81.42 (-1.07)	78.86 (-3.63)
	COMET Kiwi	80.66	77.73 (-2.93)	80.69 (+2.96)	80.69 (+0.03)	76.91 (-0.82)

Table 3: Performance scores of the Multi-agentMT system for English–X directions. Initial translations (*Translate*) are produced by Gemini-1.5-Flash. Colored values indicate score differences from the previous stage: positive and negative. For TER, variations are shown in black, as they do not directly indicate positive or negative changes.

Language	Trans-PE	PE-PR	Trans-PR
En-Icelandic	13.12	29.58	31.89
En-Serbian	13.69	31.28	33.43

Table 4: Edit distance measured by TER between stages in the Multi-agentMT workflow. 'Trans', 'PE', and 'PR' denote the Translation, Post-edit, and Proofread agents, respectively.

Table 4, the largest changes occur between Postedit and Proofread (PE–PR), approximately 2.25× greater than between Translate and Postedit (Trans–PE). When comparing Translate and Proofread (Trans–PR), about 33.3% of edits are introduced, indicating that **the final output of the TPP workflow diverges substantially from both the initial translation and the post-edited version.** Moreover, the English–Serbian pair exhibits more edits than English–Icelandic, suggesting a possible link between a higher volume of edits and stronger performance (see Figure 3).

Taken together, these results suggest that **the model primarily performs phrase-level modifications while preserving overall structure, and that encouraging more edits can improve translation quality when Postedit Agent is involved**. In this regard, our strategy of discouraging "no-error" responses appears effective, as reflected in the steadily increasing TER scores across stages. Ultimately, however, determining the benefit of these changes requires human evaluation.

5.2 Qualitative Study

This section provides qualitative examples of the Multi-agentMT framework to illustrate its operational behavior. Due to space limitations, additional examples are included in the Appendix. The example in Table 5 demonstrates a case where the Postedit Agent produces a suboptimal output, but the Proofread Agent subsequently corrects the error. **A key feature of Multi-agentMT is that the Postedit Agent can identify revision points even when its own edits lead to incorrect translations, a behavior not typically observed in single-step large models.** In this case, the Postedit Agent retained the source term "*blast*," which the Proofread Agent revised by modifying its surrounding context.

However, the Proofread Agent also shows a tendency to hallucinate by omitting portions of the input when processing longer sentences, thereby disregarding document-level boundaries. As shown in Table 6, approximately half of the content is missing from the Proofread Agent's output. Such omissions occur relatively frequently with long sentences, and warrant further investigation in future work.

6 Conclusion

We presented the potential of an AI Agent workflow based on Translate–Postedit–Proofread with a lightweight LLM, submitted as our primary system to the WMT25 General Shared Task. Official results indicate that the model is promising in low-resource settings, outperforming systems not specifically trained for such languages. Our experiments further show that the Postedit Agent plays a central role in introducing semantic-level revisions and mitigating hallucinations. Under the hypothesis that comparable quality to large models such as GPT-4o can be achieved, the workflow reduces cost to roughly half. A definitive conclusion, however, requires validation through human evaluation.

Acknowledgment

This research was supported by G-LAMP Program of the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education (No. RS-2025-25441317)."

References

Vincent Briva-Iglesias. 2025. Are ai agents the new machine translation frontier? challenges and opportunities of single-and multi-agent systems for multilingual digital communication. *arXiv preprint arXiv:2504.12891*.

Daniel Deutsch, Eleni Briakou, Isaac Caswell, Max Finkelstein, Roni Galor, and 1 others. 2025. WMT24++: Expanding the language coverage of wmt24 to 55 languages & dialects. *arXiv preprint arXiv:2502.12404*.

Ahrii Kim. 2025. RUBRIC-MQM : Span-level LLM-as-judge in machine translation for high-end models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 147–165, Vienna, Austria. Association for Computational Linguistics.

Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakougna, Jessica M. Lundin, Christof Monz, Kenton Murray, and 10 others. 2025a. Findings of the wmt25 general machine translation shared task: Time to stop evaluating on easy test sets. In *Proceedings of the Tenth Conference on Machine Translation*, China. Association for Computational Linguistics.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Natalia Fedorova, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakougna, Jessica Lundin, Kenton Murray, Masaaki Nagata, and 9 others. 2025b. Preliminary ranking of wmt25 general machine translation systems. *Proceedings of the Tenth Conference on Machine Translation*.

Tom Kocmi and Christian Federmann. 2023. Gemba-mqm: Detecting translation quality error spans with gpt-4. In *Proceedings of the Eighth Conference on Machine Translation*, Singapore. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2023. GPT-4 Technical Report. *arXiv preprint*.

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

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395. ACL.

Matt Post. 2018. A call for clarity in reporting bleu scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191. ACL.

Ricardo Rei, José GC De Souza, Daniel Alves, Chrysoula Zerva, Alon Farinha, and Alon Lavie. 2022. Comet-22: Unbabel-ist 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 911–918. ACL.

Ricardo Rei, Alon Lavie Farinha, Luisa Coheur, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL.

Stuart Russell and Peter Norvig. 2010. *Artificial Intelligence: A Modern Approach*, 3rd edition. Prentice Hall, Upper Saddle River, NJ.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231. AMTA.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Appendix

Example	Icelandic Translation	English Back-translation
Translate	Gagnrýnendur létu SEC-stofnunina hafa það á miðvikudagskvöld.	Critics let the SEC institution have it on Wednesday evening.
Postedit	Gagnrýnendur létu SEC blasta miðvikudagskvöld.	Critics blasted the SEC on Wednesday evening.
Proofread	Gagnrýnendur gagnrýndu SEC að kvöldi miðvikudags.	Critics criticized the SEC on Wednesday evening.

Table 5: Example output from Multi-agentMT. Modifications are highlighted in blue. The source segment is **"Critics blasted the SEC on Wednesday night."**

Example	Icelandic Translation	English Back-translation
Translate	Með því að lög um lífskostnað lýkur 31. mars 2024, leitast nýtt samráðsgjörningur við að framlengja stjórn á hækkun leiguverðs á komandi ári. Samráðið lýkur þennan mánudag og leggur til að leiga sé sett eftir lægsta þriggja samanburðarþátta: opinberu markaðsleigu; tillögu leigusalans að nýrri leigu; og „eðlilegri " hækkun sem unnin er úr nýju skammtakerfi með samanburði við prósentu af markaðsleigu.	With the Cost-of-Living Act expiring on March 31, 2024, a new consultation initiative seeks to extend control over rent increases in the coming year. The consultation ends this Monday and proposes that rent be set according to the lowest of three reference points: the official market rent; the landlord's proposed new rent; and a 'reasonable' increase calculated through a new adjustment system based on a percentage of the market rent.
Postedit	Með því að lög um lífskostnað lýkur 31. mars 2024, leitast nýtt samráðsgagn við að framlengja stjórn á hækkun leiguverðs á komandi ári. Samráðið lýkur þennan mánudag og leggur til að leiga sé sett eftir lægsta þriggja samanburðarþátta: opins markaðsleigu; tillögu leigusalans að nýrri leigu; og „eðlilegri" hækkun sem unnin er úr nýju skammtakerfi með samanburði við prósentu af markaðsleigu.	With the Cost-of-Living Act set to expire on March 31, 2024, a new consultation document aims to extend control over rent increases in the coming year. The consultation ends this Monday and proposes that rent be determined based on the lowest of three reference points: open market rent; the landlord's proposed new rent; and a 'reasonable' increase calculated using a new adjustment system that compares a percentage of the market rent.
Proofread	Með því að lög um lífskostnað rennur út 31. mars 2024, leitast nýtt samráðsgagn við að framlengja stjórn á því hversu miklar leiguhækkanir má gera á komandi ári.	With the Cost-of-Living Act set to expire on March 31, 2024, a new consultation document seeks to extend control over how much rent can be increased in the coming year.

Table 6: Example of hallucination produced by the Proofread Agent. Modifications are highlighted in blue. The source segment is **"With the Cost of Living Act legislation ending on 31 March 2024, a new consultation document seeks to extend controls on the level of rent increases that can be levied in the coming year. The consultation ends this Monday and proposes rents be set by the lowest of three comparators: open market rent; a landlord's proposed new rent; and a "reasonable" increase devised from a new taper system using comparison with a percentage of market rent."**

Listing 1: Prompt of Postedit Agent. The use of reference is optional.

```
{source language} source: ```{source sentence}```
{target language} translation: ```{translation}```
(Optional) {target language} reference: ```{reference}```

Based on the source [and reference] and translation enclosed in
   triple backticks, identify only errors in the translation and
   classify each by category.
Categories: addition, mistranslation, omission, untranslated text,
   grammar, inconsistency, punctuation, word order, terminology, and
   style. You must find at least one issue, even minor, stylistic, or
    subjective.
Rate severity from 1 (minor) to 4 (severe distortion). Never select
   entire sentences or long phrases as an error span. Select only the
    exact word or short phrase where the error occurs. Suggest fixes
   *only* for the erroneous parts -- do not rewrite the full sentence
   .

Format:
{
  "<error span>": {
    "category": "<category>",
    "severity": <1-4>,
    "suggestion": "<fix>"
  },
  ...
}
```

Listing 2: Propmt of Proofread Agent

```
Review the given translation for errors. Find errors and correct them
    first. Then, generate five rephrased translations optimized for
   fluency and adequacy in the {domain} domain. Select the most
   contextually appropriate version based on linguistic fluency in {
   target language}, preservation of source accuracy, and adherence
   to professional translation standards. Output only the final best
   translation. Do not include the other versions, reasoning, or any
   additional text. The output must consist of a single sentence only
   .

{source language} source: ```{source sentence}```
{target language} translation: ```{translation}```
```